

---

# Unit 5/5: RNA-seq analysis

## RNA-seq bioinfo analysis

Oct. 3rd, 4th, 5th, & 6th, 2023

Camille Marchet - Pierre Pericard

---

---

# Day 2 & 3 : RNA-seq bioinfo analysis

## Day 2

- Lecture: with reference RNA-seq
  - RNA-seq QC + Cleaning
  - Mapping on reference
  - Assembly with reference
  - Quantifying gene expression
- Practical: with reference and de-novo RNA-seq (Day 2 & 3)

## Day 3

- Lecture: *de-novo* RNA-seq
  - de-novo assembly
  - local assembly for variant calling
- Practical: with reference and de-novo RNA-seq (Day 2 & 3)
- Lecture: Introduction to long-reads RNA-seq

# General Introduction

# Goals

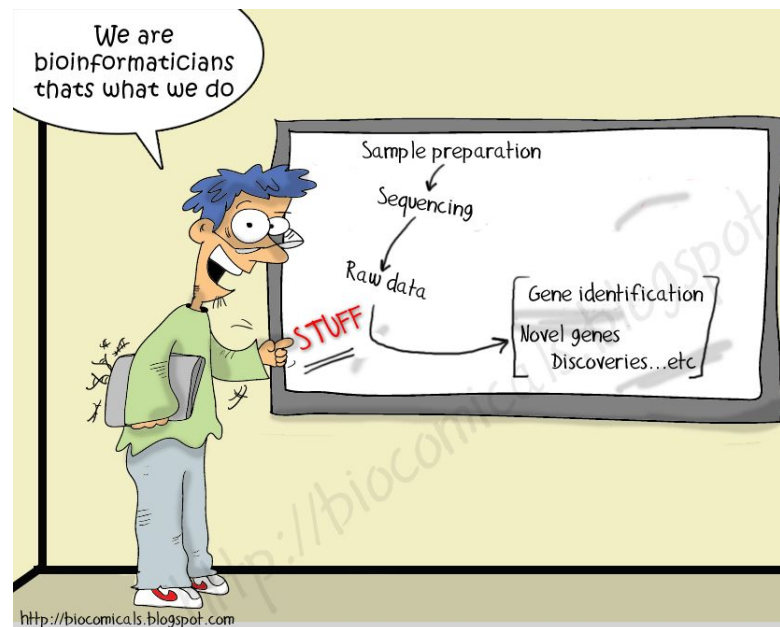
This course main goals:

- An overview of RNA-seq data analysis
- Identify the (key issues/points) (critical steps/parameters)

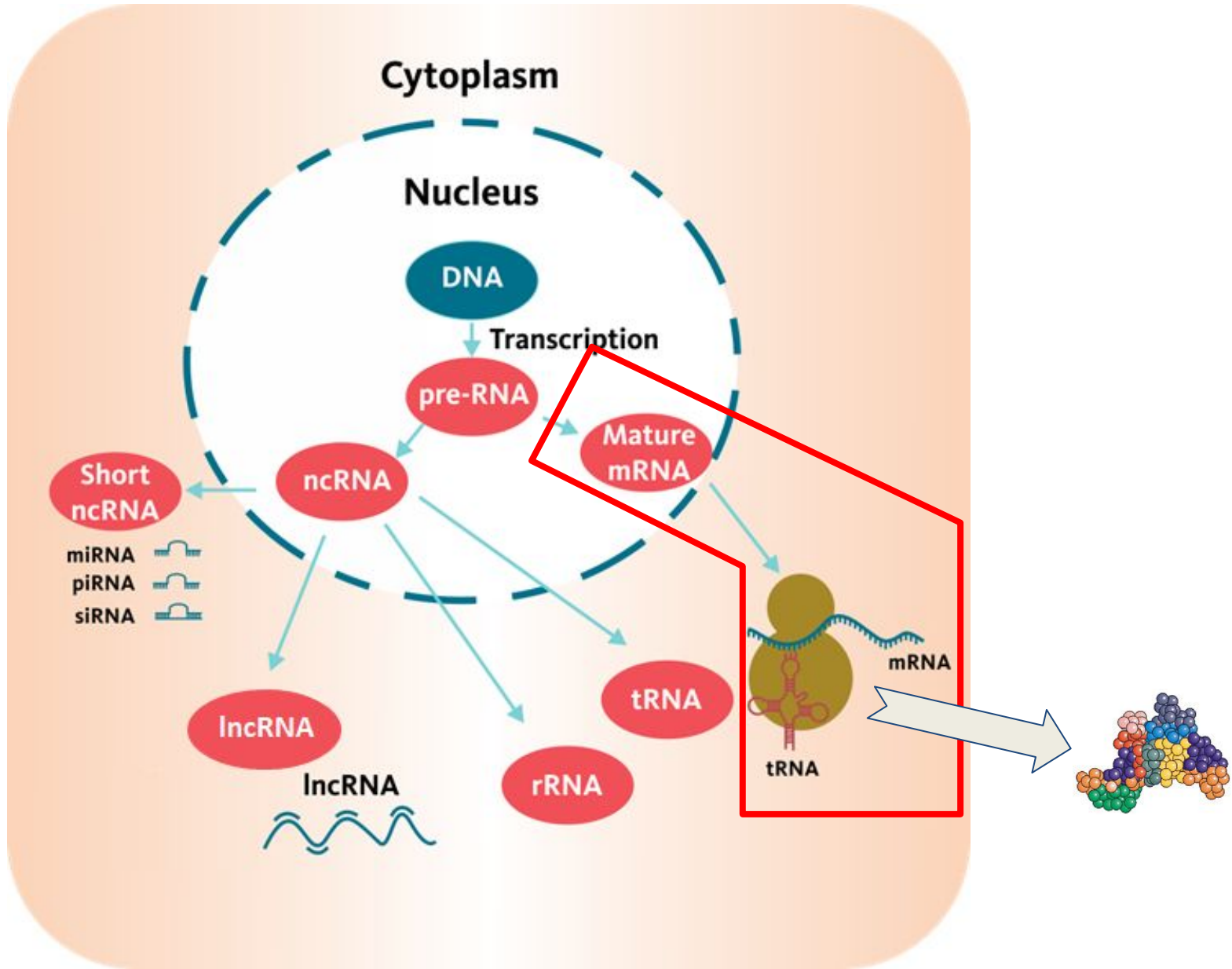
# Warning !

This is NOT a course to train you as a bioinformatician, and this course will NOT allow you to design an analysis pipeline set-up for your specific needs

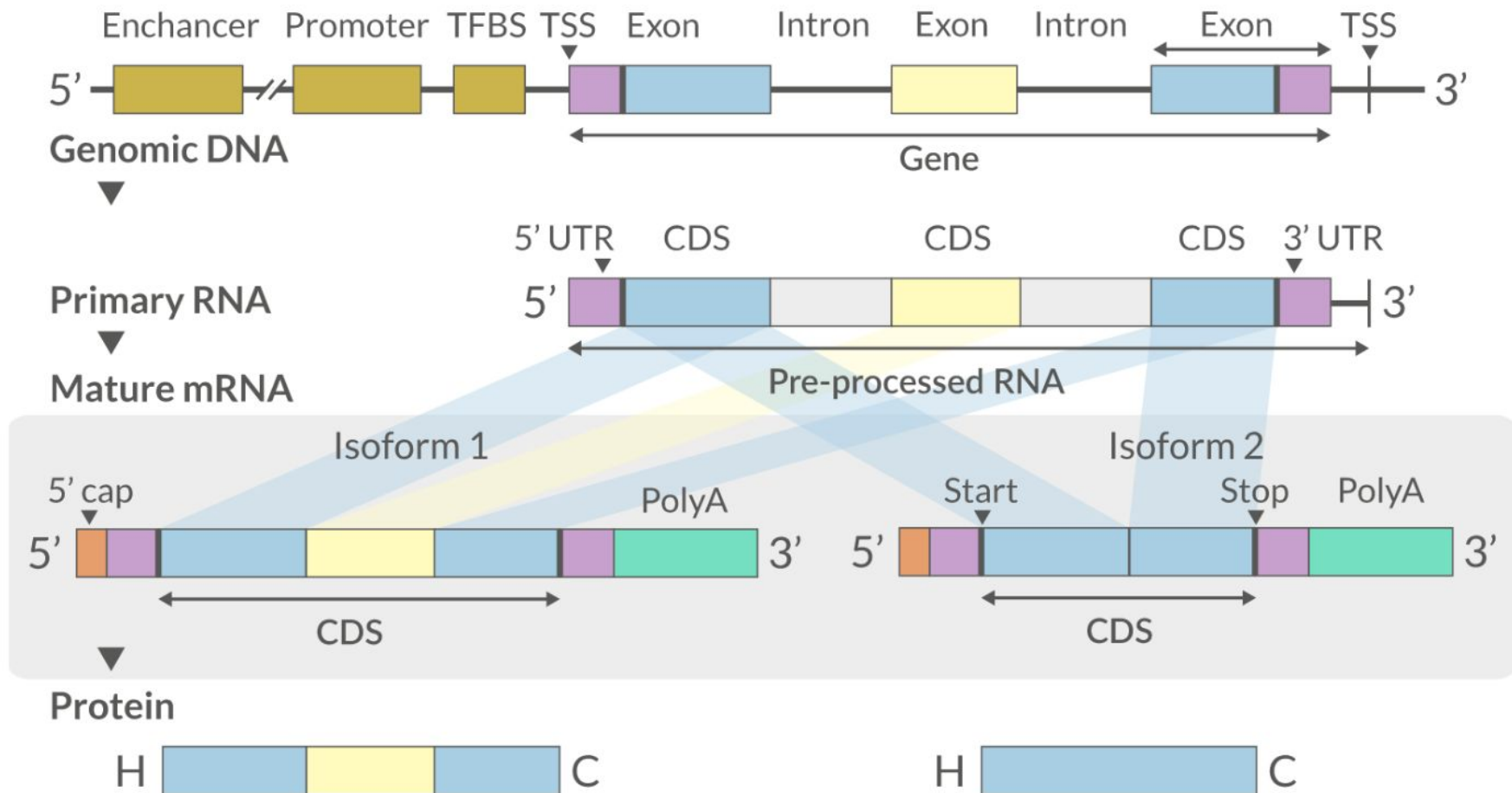
This course WILL give you the basis information to understand and run a generic RNA-seq analysis, its key steps and problematics, and how to interact with bioinformaticians/bioanalysts that can analyze your RNA-seq datasets



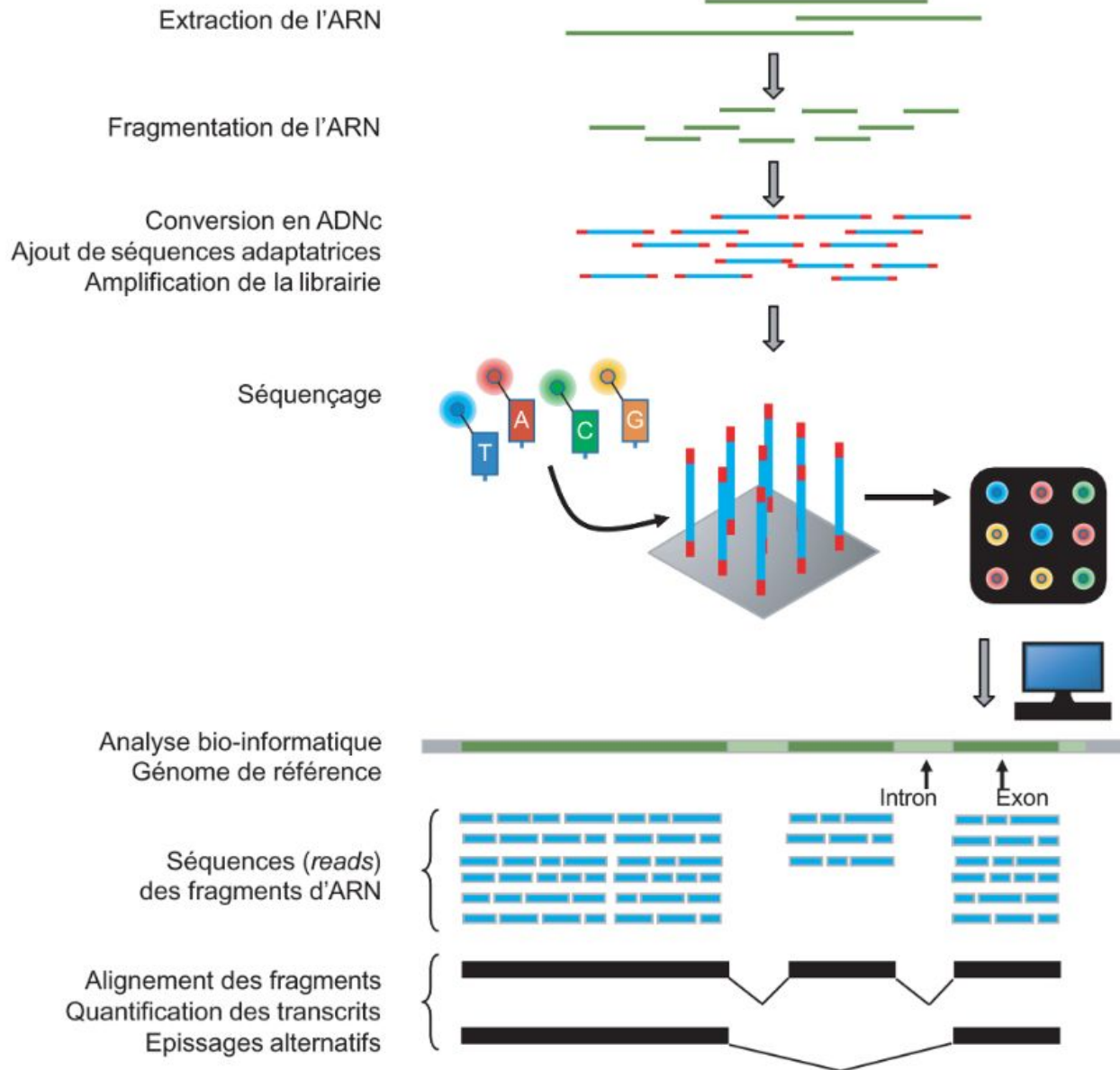
# Preliminary



# Maturation and variability of RNA



# Sequencing overview





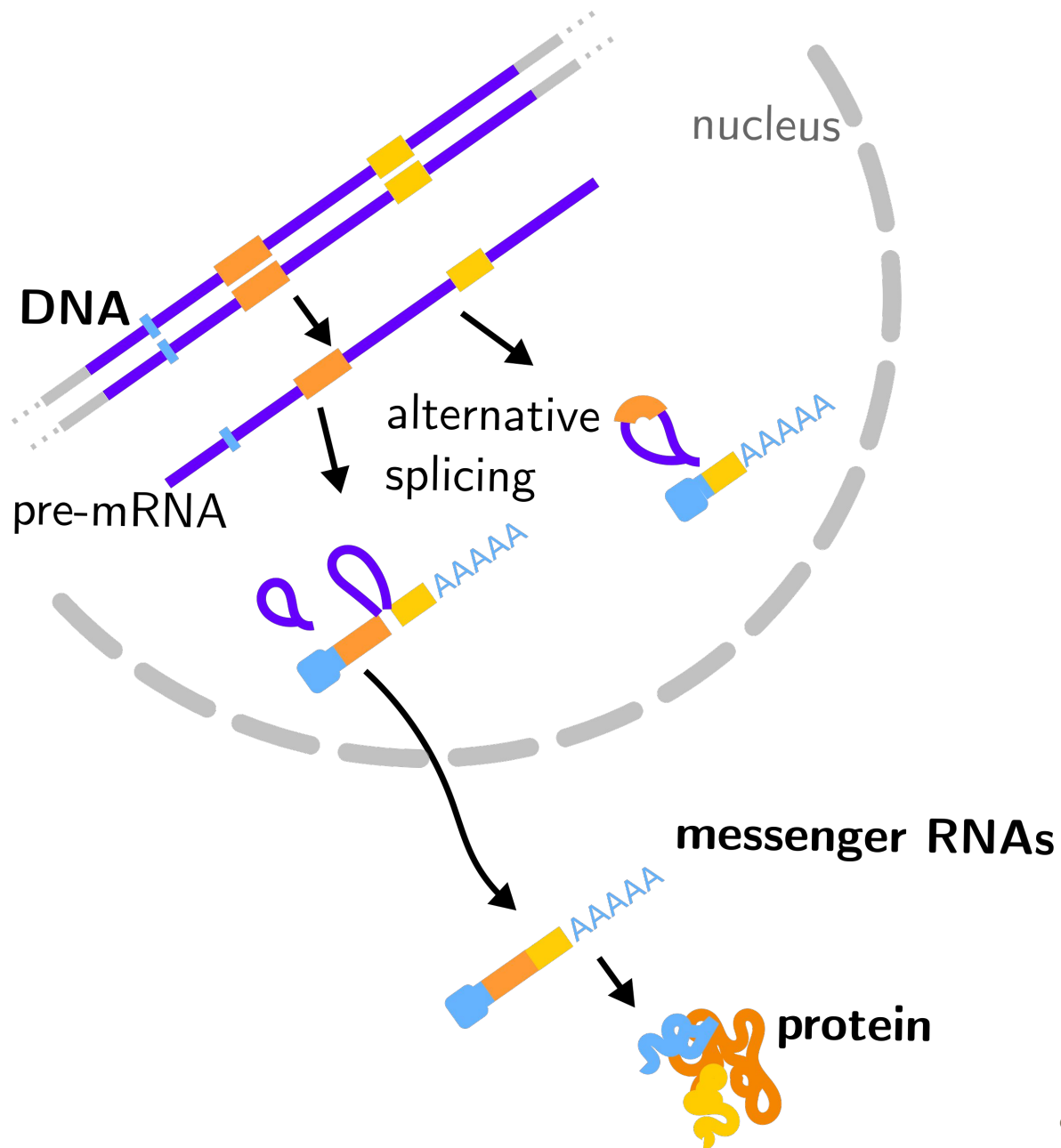
# Preliminary

Transcriptome/transcript

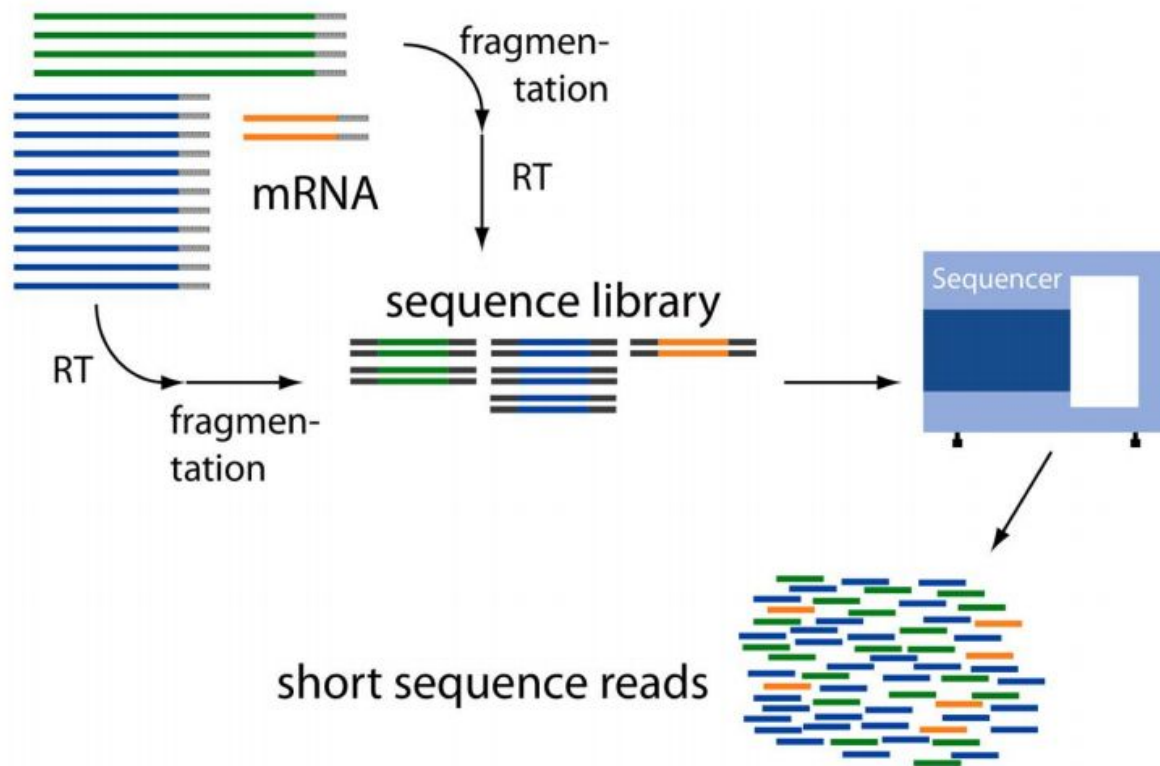
Transcriptomics

(Alternative) isoform

Splicing



# Sequencing: overview



From: <http://www2.fml.tuebingen.mpg.de/raetsch/members/research/transcriptomics.html>

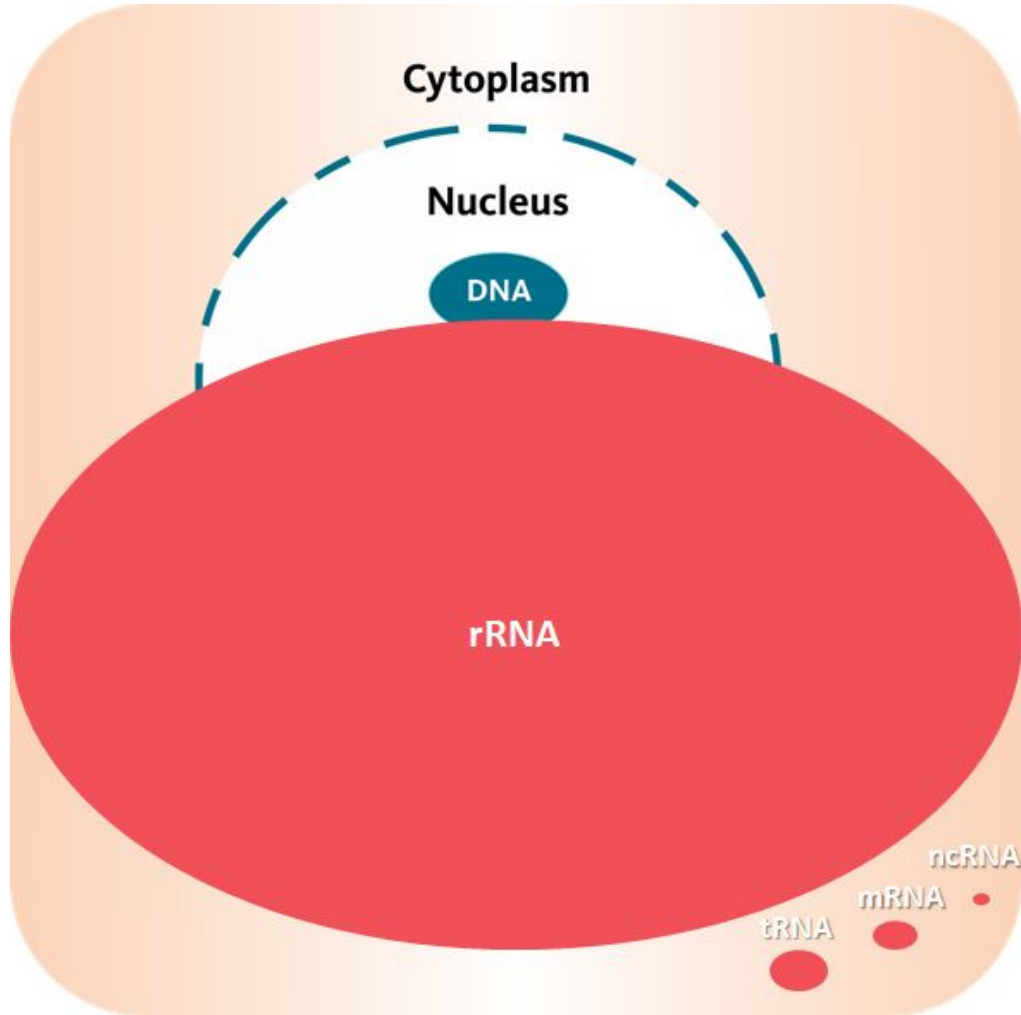
# How to make cDNA libraries

- Extract RNA, convert to cDNA
- pass to next gen sequencer
- millions to billions of reads

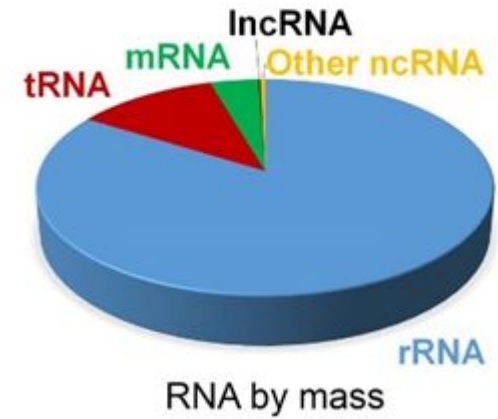
make cDNA?

- Prime mRNA with random hexamers R6
  - reverse transcriptase => cDNA first strand synthesis
  - then second strand
- => illumina cDNA library

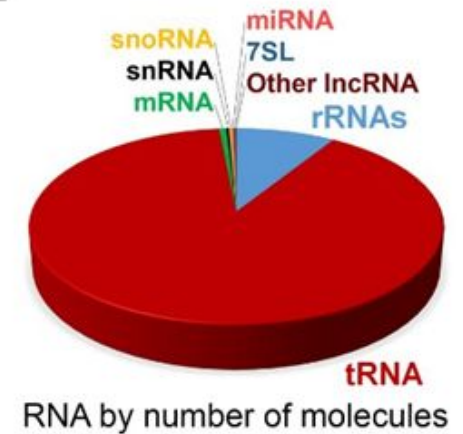
# Do you want *all* RNAs?



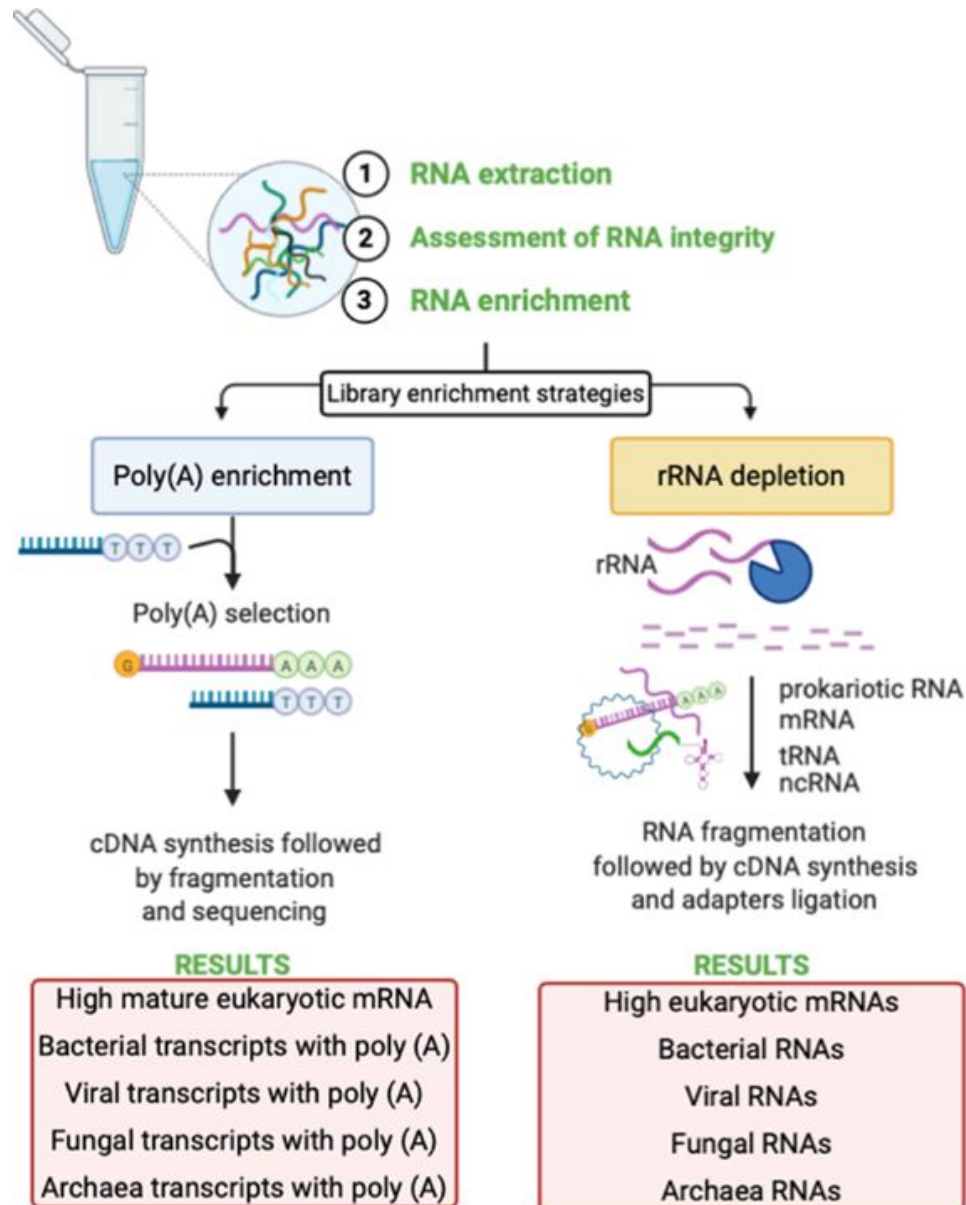
A



B



# How to make cDNA libraries



# How to sequence (1)

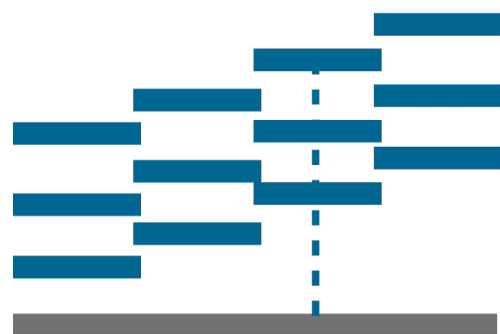
## RNA selection

- polyA+
- Ribo-Zero (human, mouse, plants, bacteria, ...)  
  
(ARN = 90% of ARNr, 1-2% of ARNm)
- in prokaryotes: no polyA (= no capture), no splicing (= less complex)

## Types of reads, experimental design

- paired-end
- replicates

## How to sequence (2)



# RNA-seq

- reads around **150-200** bp
- the number of **detected transcripts increases with the sequencing depth**
- the **expression** measure is **more precise with more depth**
- 5 million reads can be enough to detect genes mildly-highly expressed in human
- **30 million reads: a typical middle-size project for quantification**
- 100 million must be preferred to detect lowly expressed genes (see for instance **saturation curves** in “Differential expression in RNA-seq: a matter of depth.” *Genome Res.* 2011)
- these numbers depends on the species/tissues (complex splicing...)
- keep **replicates** in mind



# There are plenty of protocols...

Méthode	Description	Référence
mRNA-seq	Identification les ARN messagers.	[Mortazavi et al., 2008]
miRNA-seq	Identification les micro ARN.	[Ruby et al., 2006]
GRO-Seq (Global Run-On Sequencing), PRO-Seq (Precision Run-On Sequencing) et NET-Seq (Native elongation transcript sequencing)	Sélection et séquençage uniquement les ARNs en cours de transcription par l'ARN polymérase II.	[Core et al., 2008] [Kwak et al., 2013] [Churchman and Weissman, 2011]
Ribo-Seq (Ribosome profile sequencing) et TRAP-Seq (Targeted purification of polysomal mRNA sequencing)	Identification les ARNs messagers en cours de traduction.	[Ingolia et al., 2009] [Reynoso et al., 2015]
RIP-Seq (RNA immunoprecipitation sequencing), CLIP-Seq (Cross-linking and immunoprecipitation sequencing), PAR-CLIP (Photoactivatable-ribonucleoside-enhanced cross-linking and immunoprecipitation) et iCLIP (individual-nucleotide resolution CLIP)	Détermination des régions d'ARN liées à une protéine d'intérêt.	[Cloonan et al., 2008] [Chi et al., 2009] [Hafner et al., 2010] [Huppertz et al., 2014]
ChIRP-Seq (Chromatine isolation by RNA purification)	Identification des régions du génome qui interagissent avec l'ARN.	[Chu et al., 2011]
PARE-Seq (Parallel analysis RNA ends sequencing)	Etude des sites de clivage des micro-ARNs ainsi que de la dégradation des ARNs.	[German et al., 2009]



# Sequencing reads file formats

## FastQ

READ

1. Identifier

2. Sequence

4. Quality scores (as ASCII chars)

```
@SRR062641.6751359  
CGCCCGGCCAATCATTGTGGTTTTAAGTCACTAAGTTTGAGGCTATTTTGTTTTACAGCAAAGCTAACTGATGCAGACAGGGACAAGTCAGTCTCATCT  
+  
CBLNPGJQQQJPPQPPQPQRGPPPPRRQRPSRGRQQRLRRRMEPQQPMJHQEHEKMMFIIRH?SIIHKNJIKRLJJKIHEABHIFGCGGEFCGDGDCE
```

```
@SRR062634.16249693  
CTAAGTTTGAGGCTATTTTGTTTTACAGCAAAGCTAACTGATGCAGACAGGGACAAGTCAGTCTCATCTCTGTGCACCCAGCATTGCCCAGAACAGGGC  
+  
ALKMOOOOPPQJQOPPPPPQPPPPPPRJRQROQQQRPPRQPPQPFQSQQPRLLIMHKSJRQORMFELRPQNQRQJQRRPQQLIRKDMKQJPN8CFDGDCCCB
```

```
@SRR062634.20060465  
CTCCCAGCTTCCAACAGACCCTGTCCCAGCTCCCTCCAAGCTGAGTGTTGGCCTGATACCTACCAGTGGAGCGAGGGGAACCCGAGGACTGCCAAGGGCA  
+  
D?KMPQEPGCPQONPQIQIGR@DPERQHEKBED=HCHG8EHFDCD6<329@<:69A<6, ;<967>;=C:>AA8BBED#####
```

## FastA

```
>SRR062641.6751359  
CGCCCGGCCAATCATTGTGGTTTTAAGTCACTAAGTTTGAGGCTATTTTGTTTTACAGCAAAGCTAACTGATGCAGACAGGGACAAGTCAGTCTCATCT
```

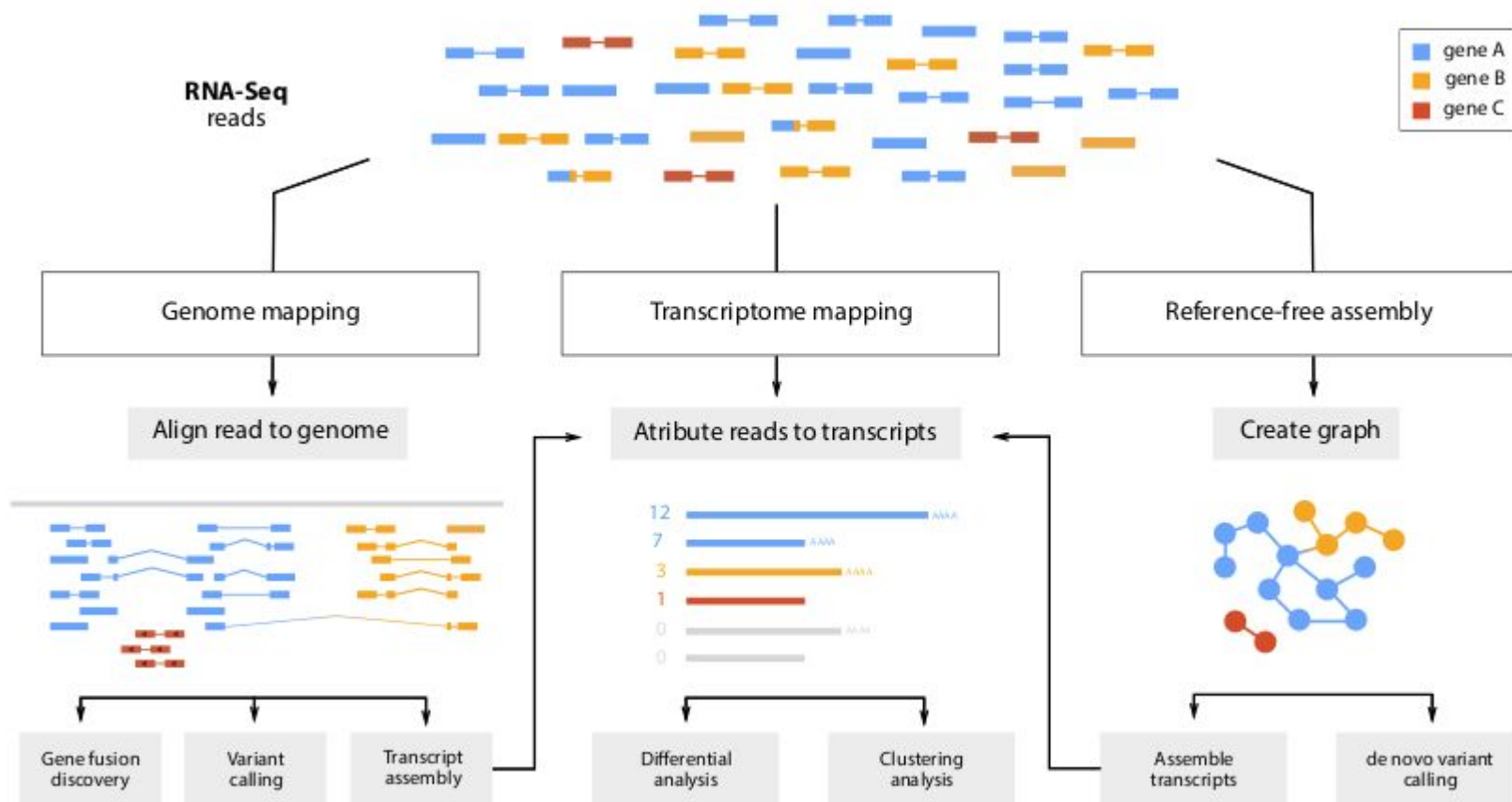
```
>SRR062634.16249693  
CTAAGTTTGAGGCTATTTTGTTTTACAGCAAAGCTAACTGATGCAGACAGGGACAAGTCAGTCTCATCTCTGTGCACCCAGCATTGCCCAGAACAGGGC
```

```
>SRR062634.20060465  
CTCCCAGCTTCCAACAGACCCTGTCCCAGCTCCCTCCAAGCTGAGTGTTGGCCTGATACCTACCAGTGGAGCGAGGGGAACCCGAGGACTGCCAAGGGCA
```

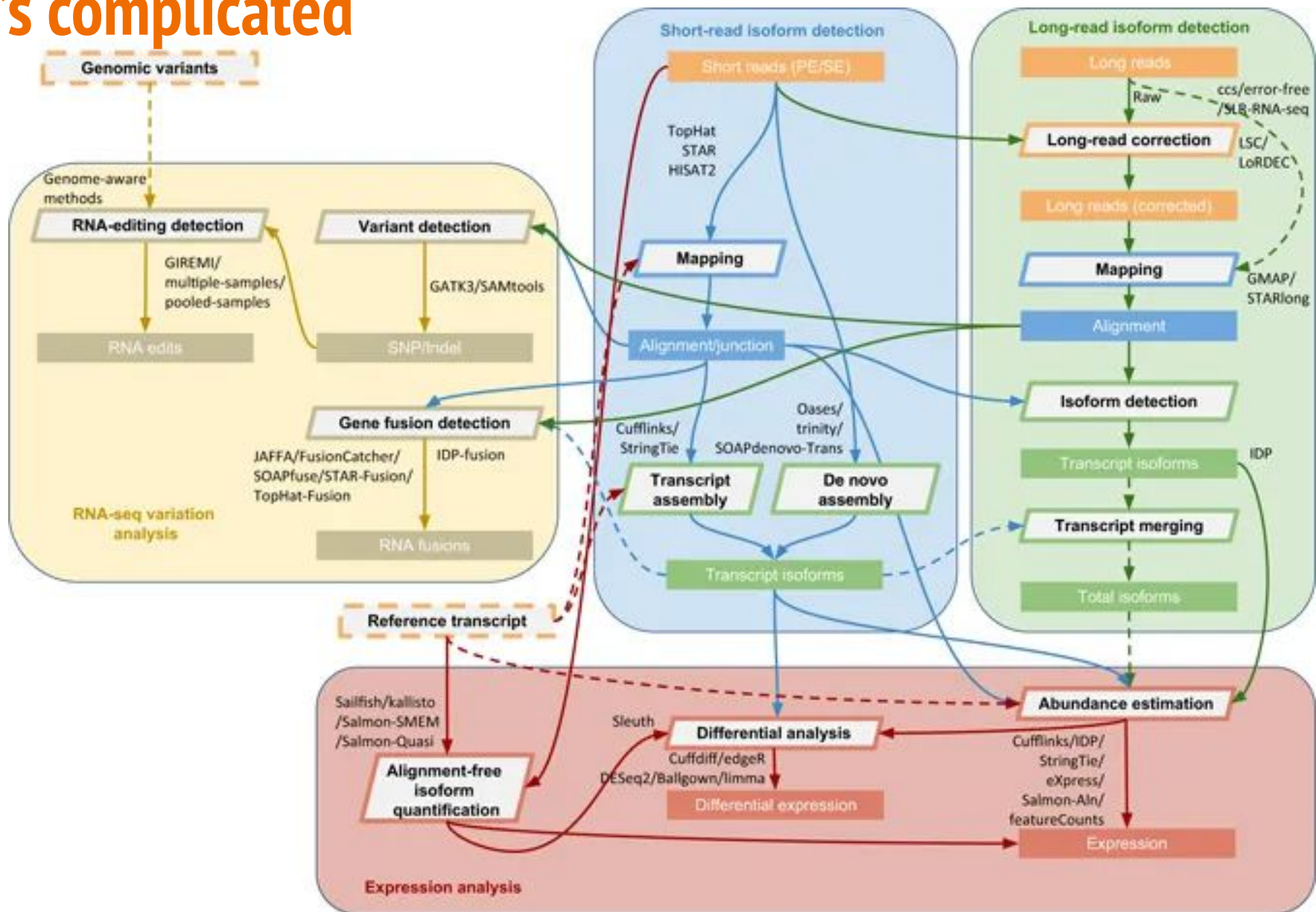
Mais aussi: FAST5, BAM, ...



# What people do with their RNA-seq



# It's complicated



# Outcomes of RNA-seq studies

- gene annotation
- protein/function prediction
- gene/splicing quantification
- isoform discovery/fusion transcripts/lncRNA...
- variant calling
- methylations
- RNA structures
- ...

# QC - Cleaning - Preprocessing



# Known biases in RNA-seq



# Known biases in RNA-seq

## Biological sample:

- presence of pre-mRNA
- 3' bias over-represented (RNA degradation)
- contaminations

## Library preparation:

- DNase fail
- rRNA depletion not effective
- pcr bias
- variable insert size (smaller than sequencing length)
- reads with no inserts

## Sequencing:

- quality drops at the end of reads

# Quality Control (QC)

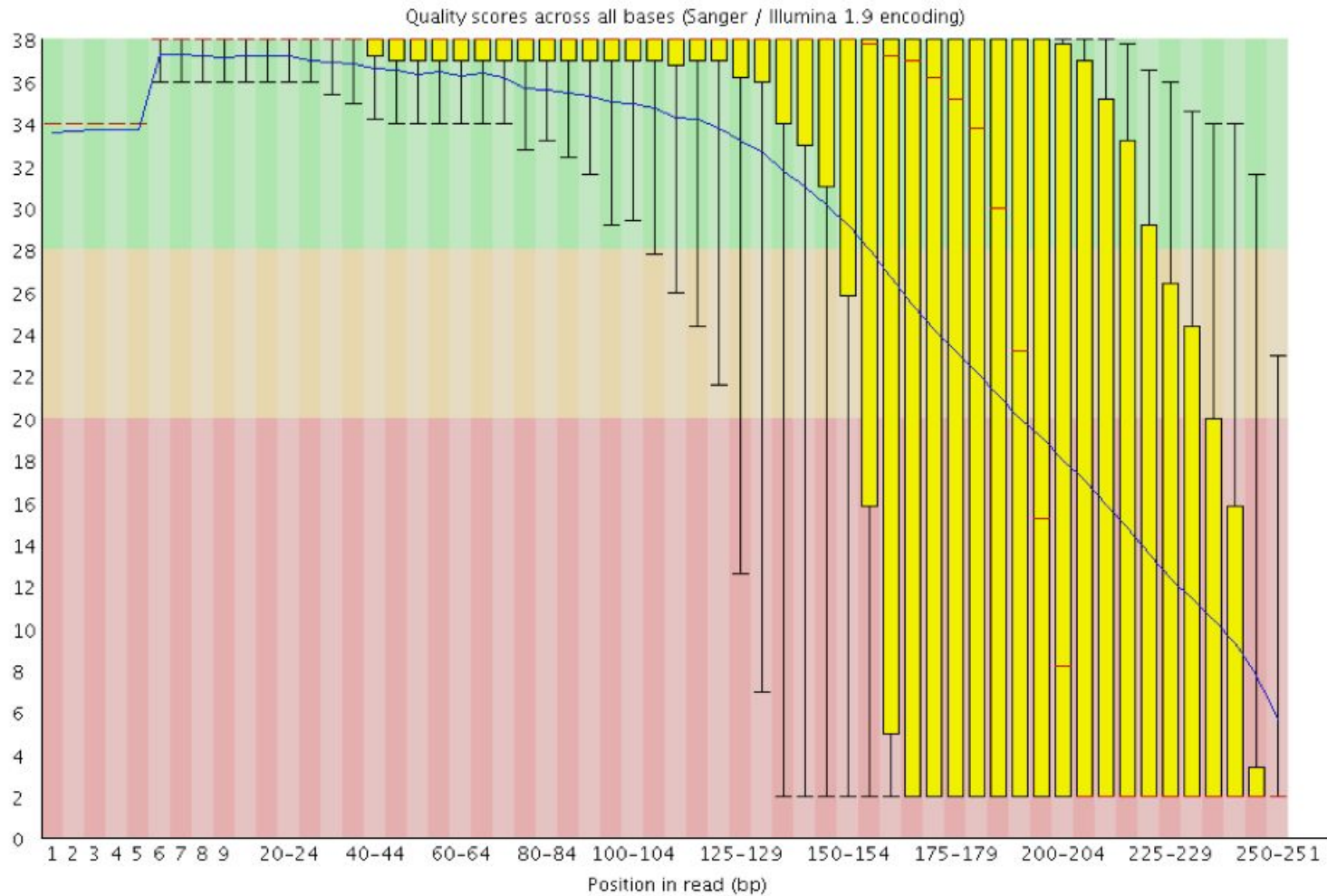
Quality Control (QC) is important to:

- Check if your sample sequencing went well
- Know when you need to sequence again (sequencing platform QC fail)
- Identify potential problems that can be fixed, or not
- Follow the impact of preprocessing steps

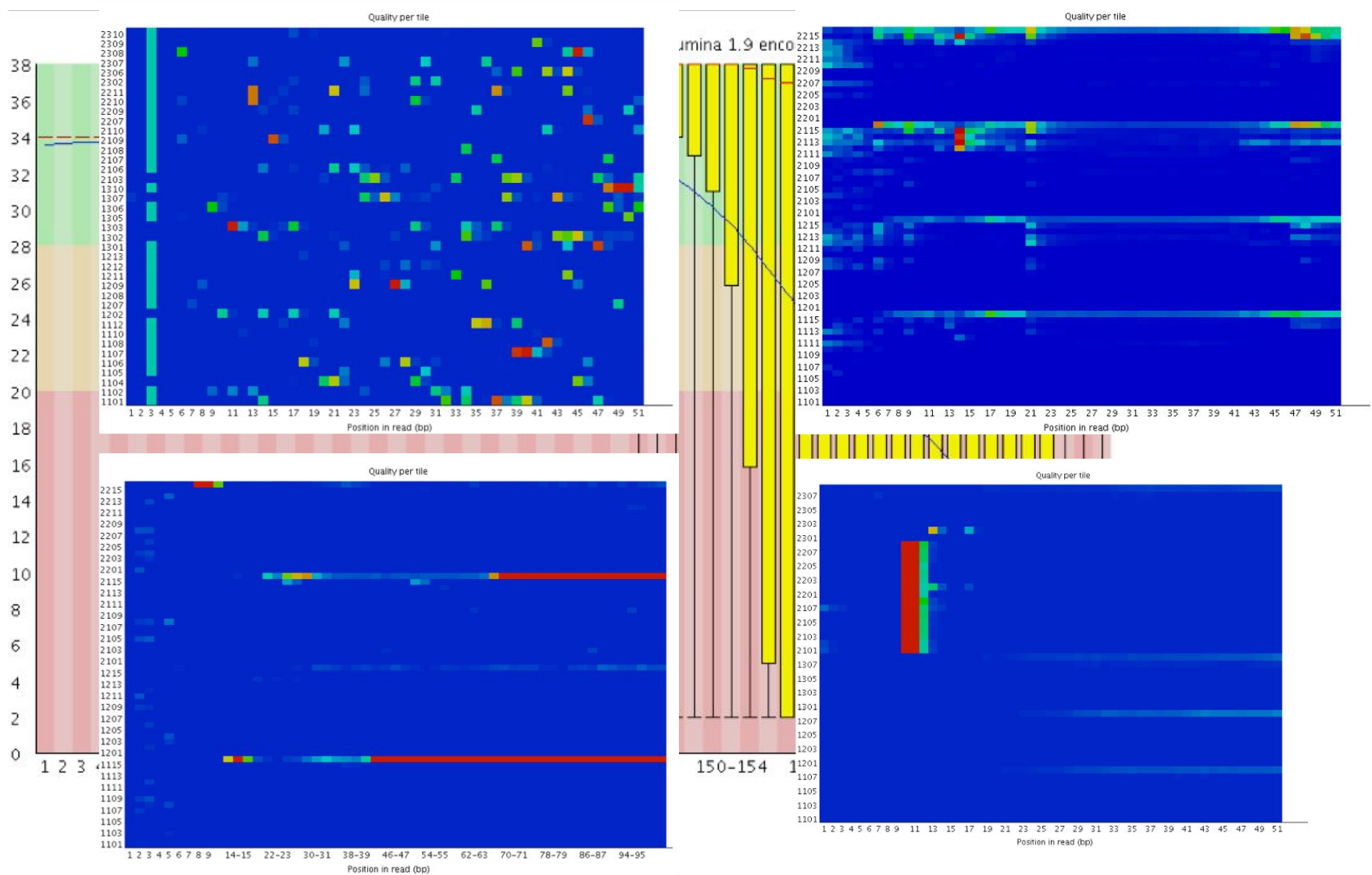
⇒ FastQC (<https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>)

+ MultiQC (<https://multiqc.info/>) when comparing multiple datasets

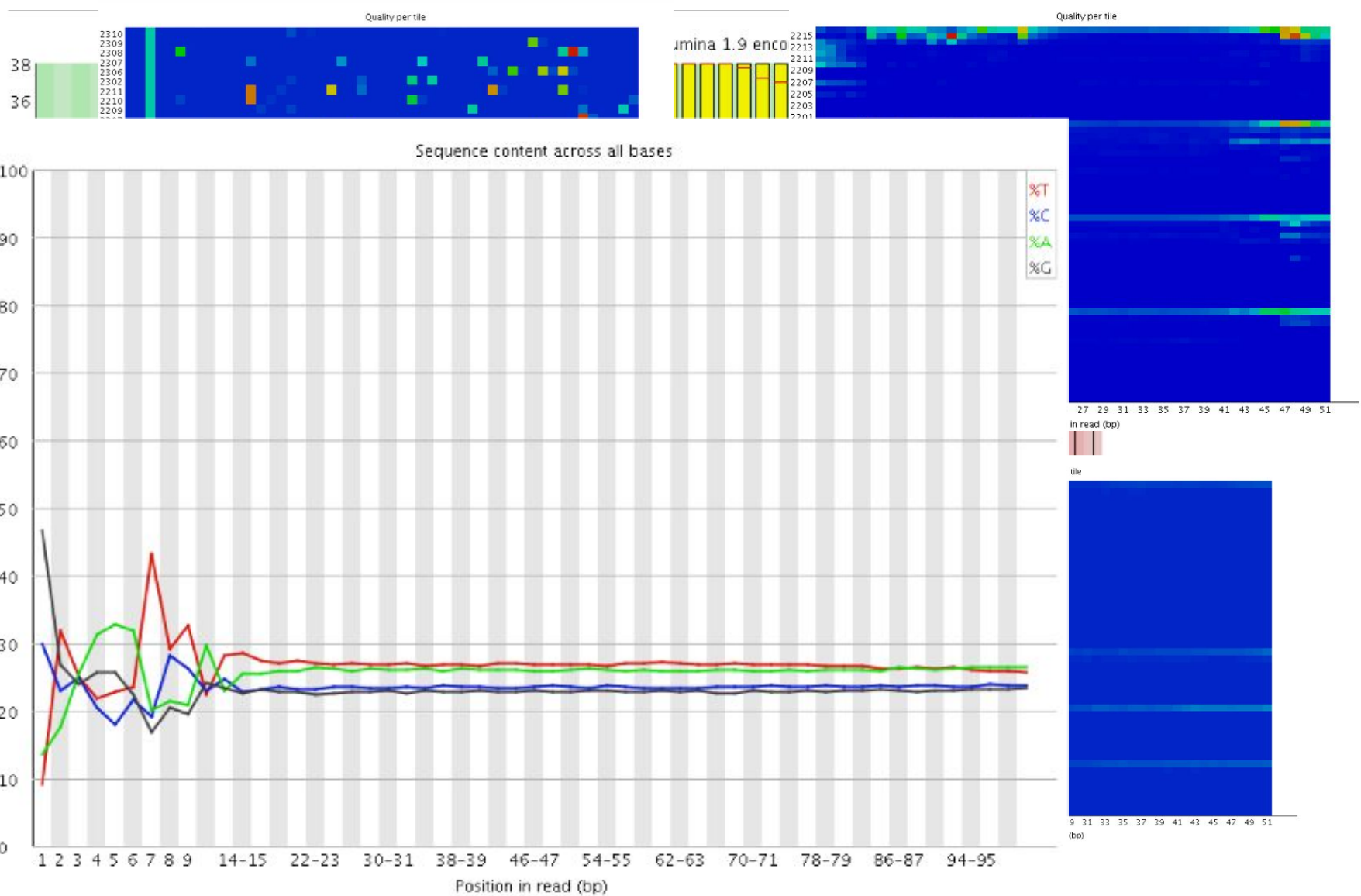
# Quality Control



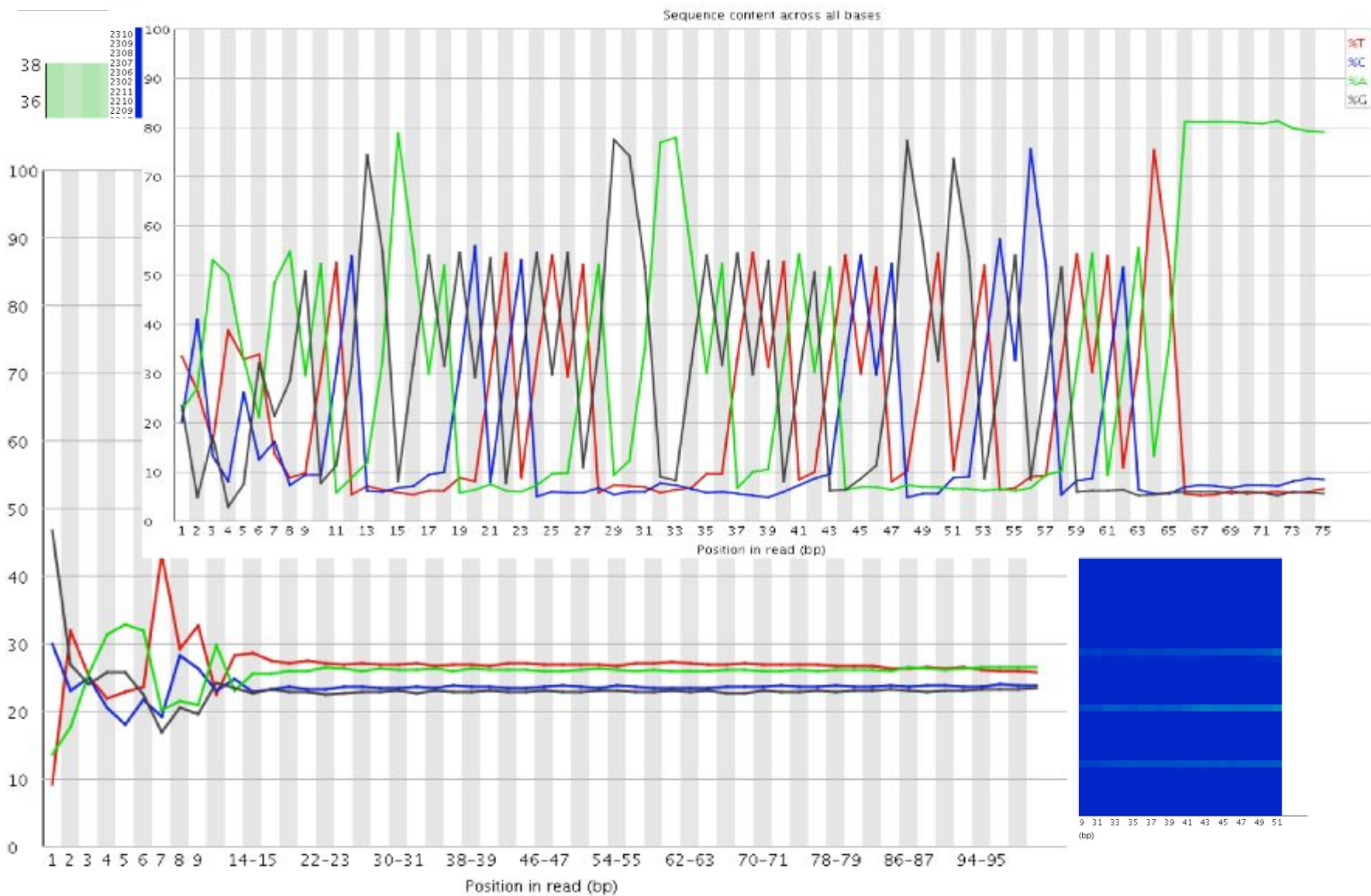
# Quality Control



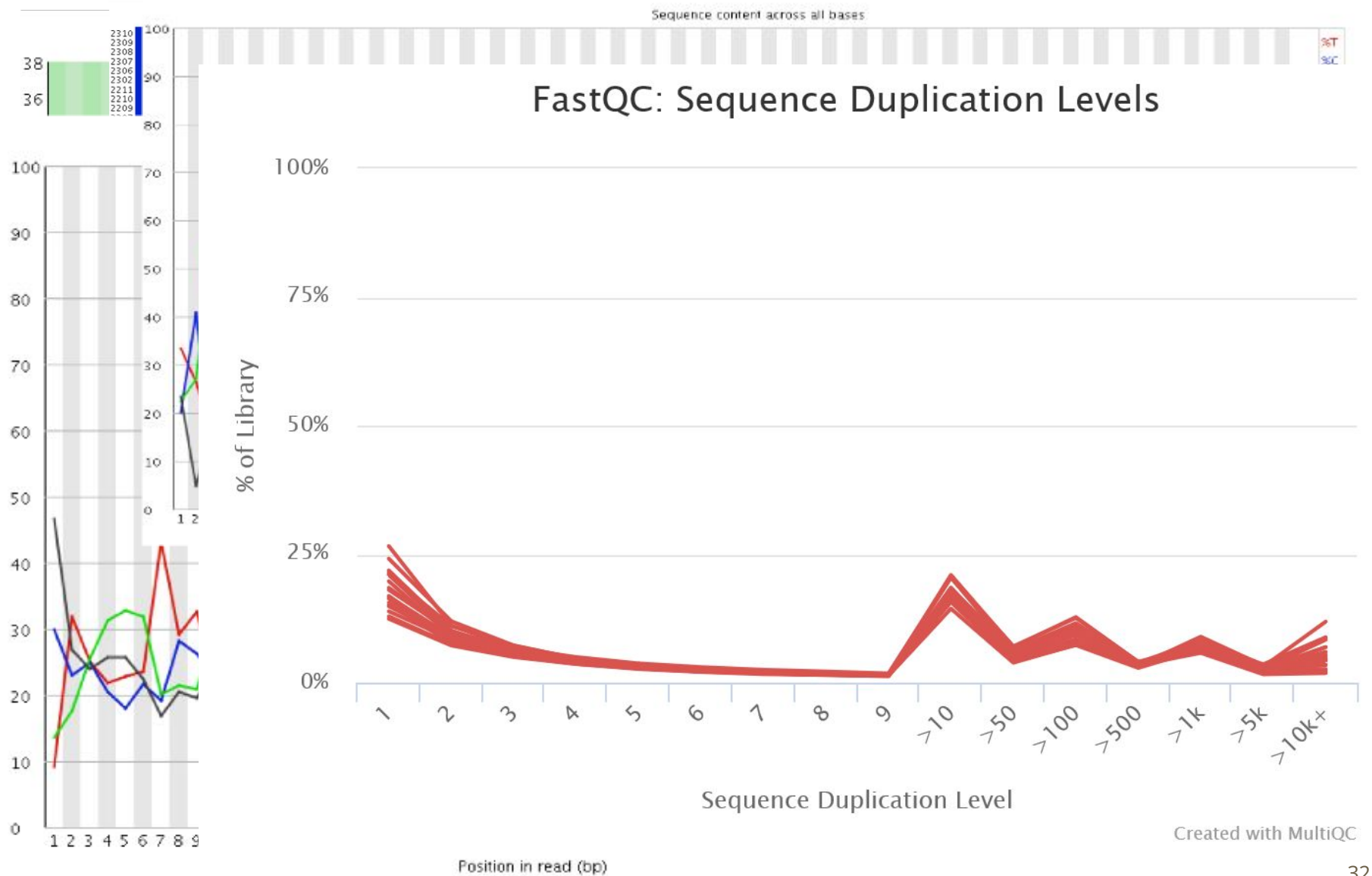
# Quality Control



# Quality Control



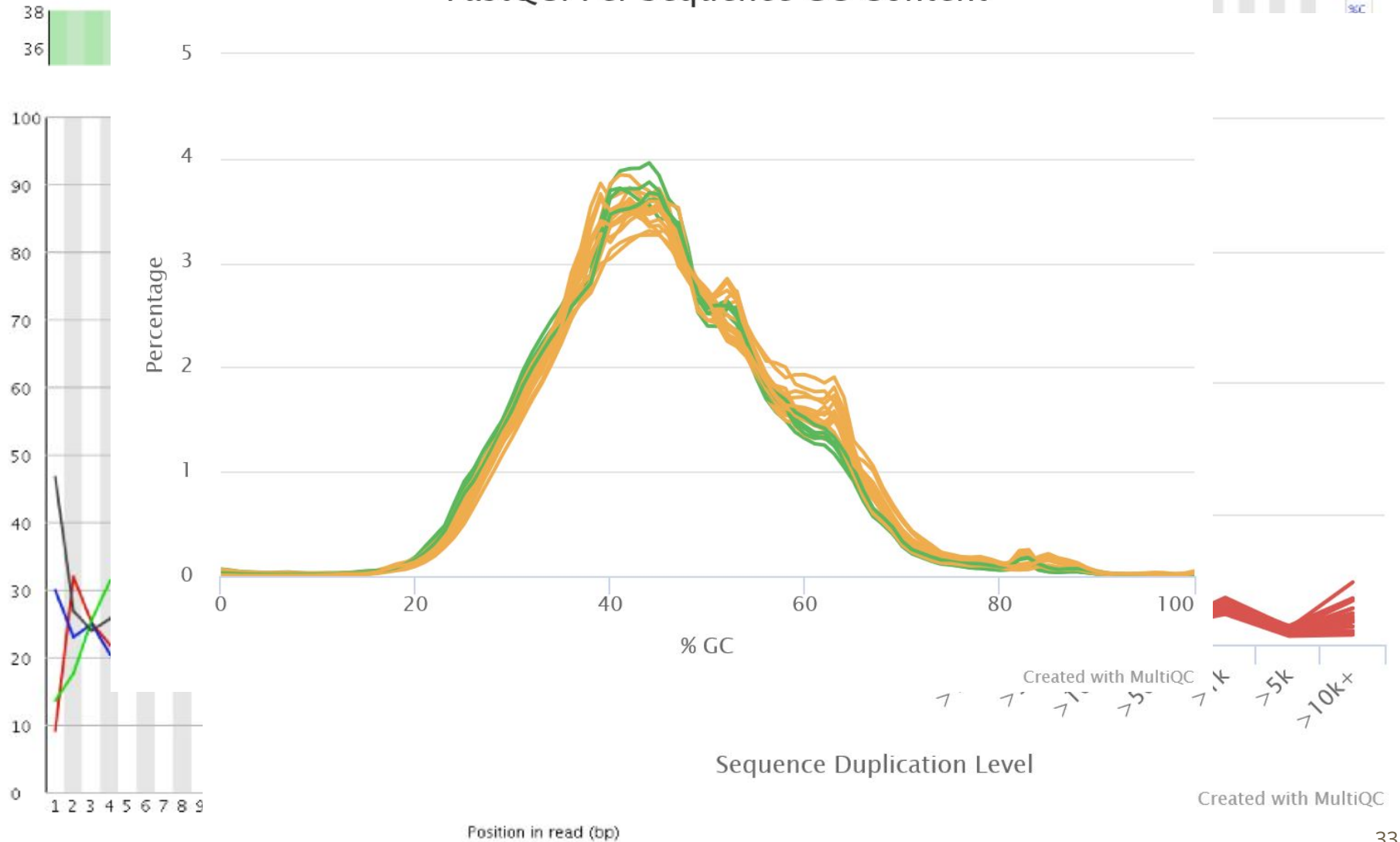
# Quality Control





# Quality Control

## FastQC: Per Sequence GC Content



# Quality Control

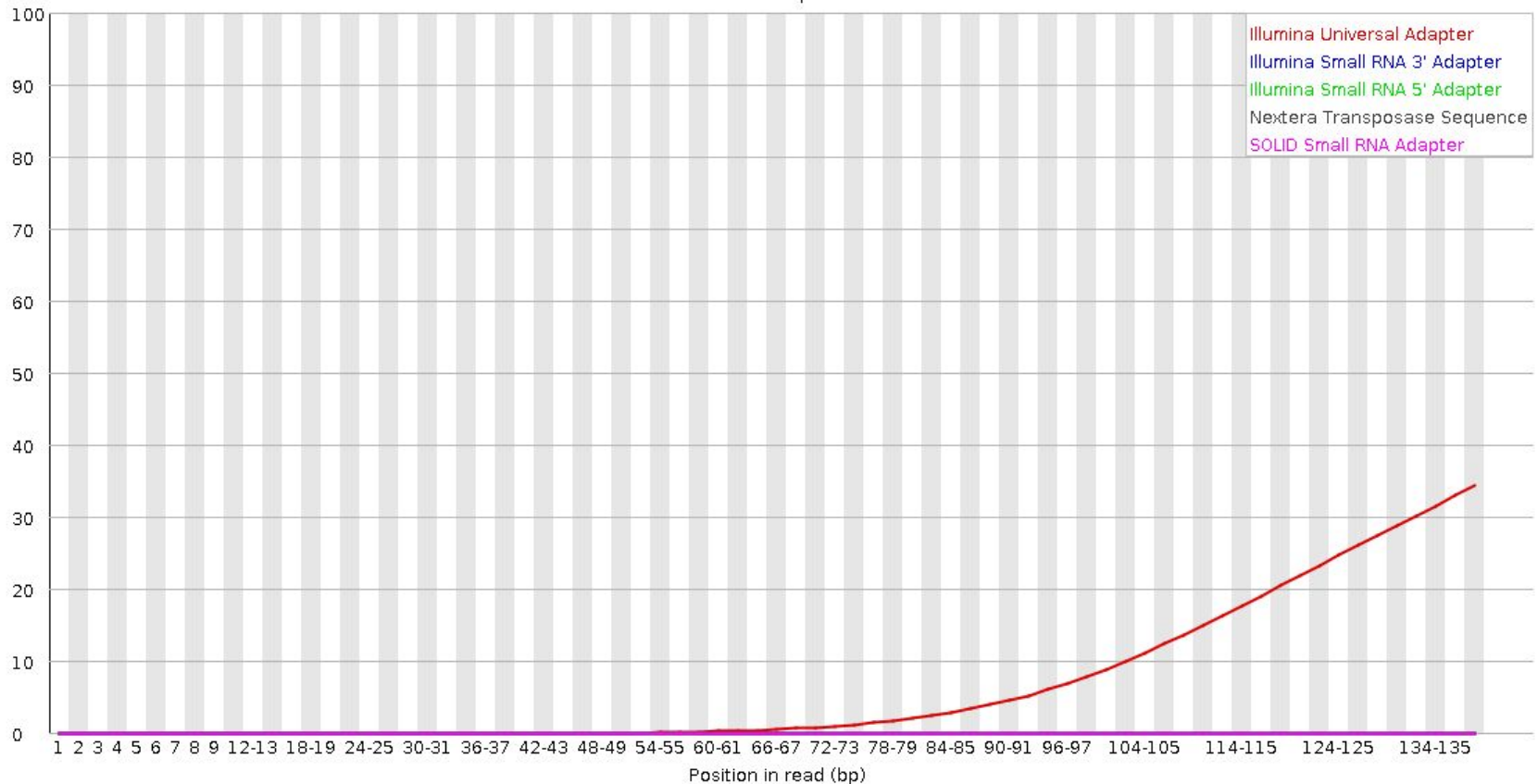
## FastQC: Per Sequence GC Content

38 1



### ✖ Adapter Content

% Adapter



1 2 3 4 5 6 7 8 9 14-15 22-23 30-31 38-39 46-47 54-55 62-63 70-71 78-79 86-87 94-95  
Position in read (bp)

# Cleaning - Preprocessing

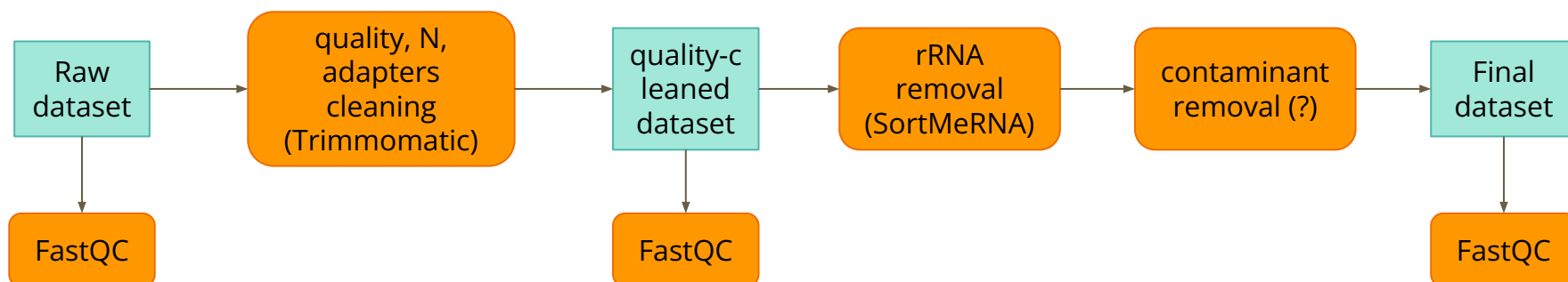
Cleaning should be done in the reverse order that errors were generated.

1. Sequencing errors: quality trimming and filtering, Ns removal
2. Library preparation: adapters removal
3. Sample contamination: rRNA, mito, other contaminants

Note 1: step 1 (quality trimming) is not considered critical anymore and could even hinder downstream tools/algorithms.

Note 2: If the reads are going to be aligned against a reference genome, this whole process can be skipped or applied very lightly

# Cleaning - Preprocessing



# Practical: Quality Control (QC) & Cleaning

Open Galaxy



Practical: NGS2023\_rnaseq\_05\_tp\_bioinfo.pdf

Shared Data → Histories → [TP RNAseq bilille Initial datasets](#)

TIAAS: <https://usegalaxy.fr/join-training/bilille-rnaseq-2023/>

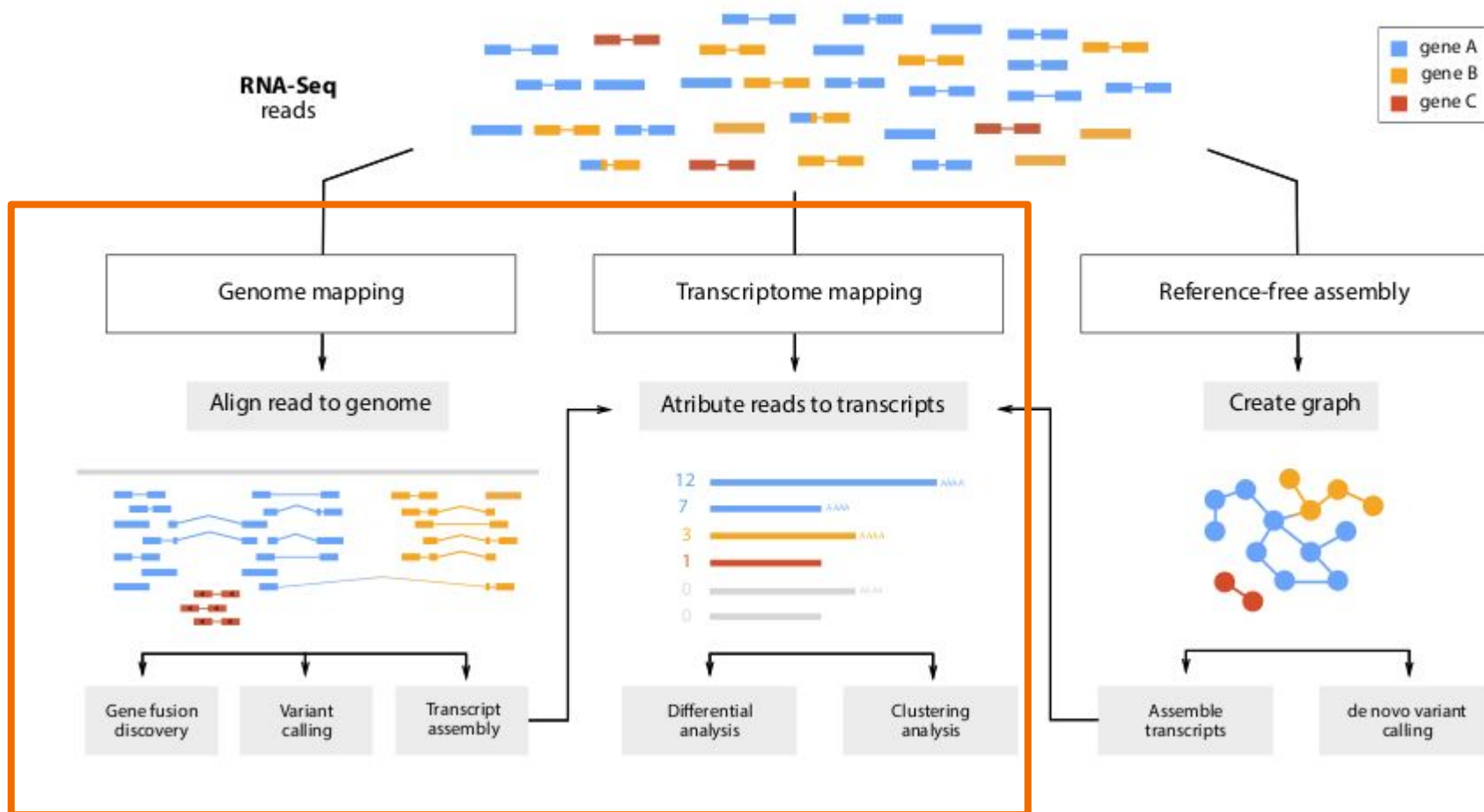
**With reference RNA-seq**

# W/ reference RNA-seq. For what purpose ?

Mainly:

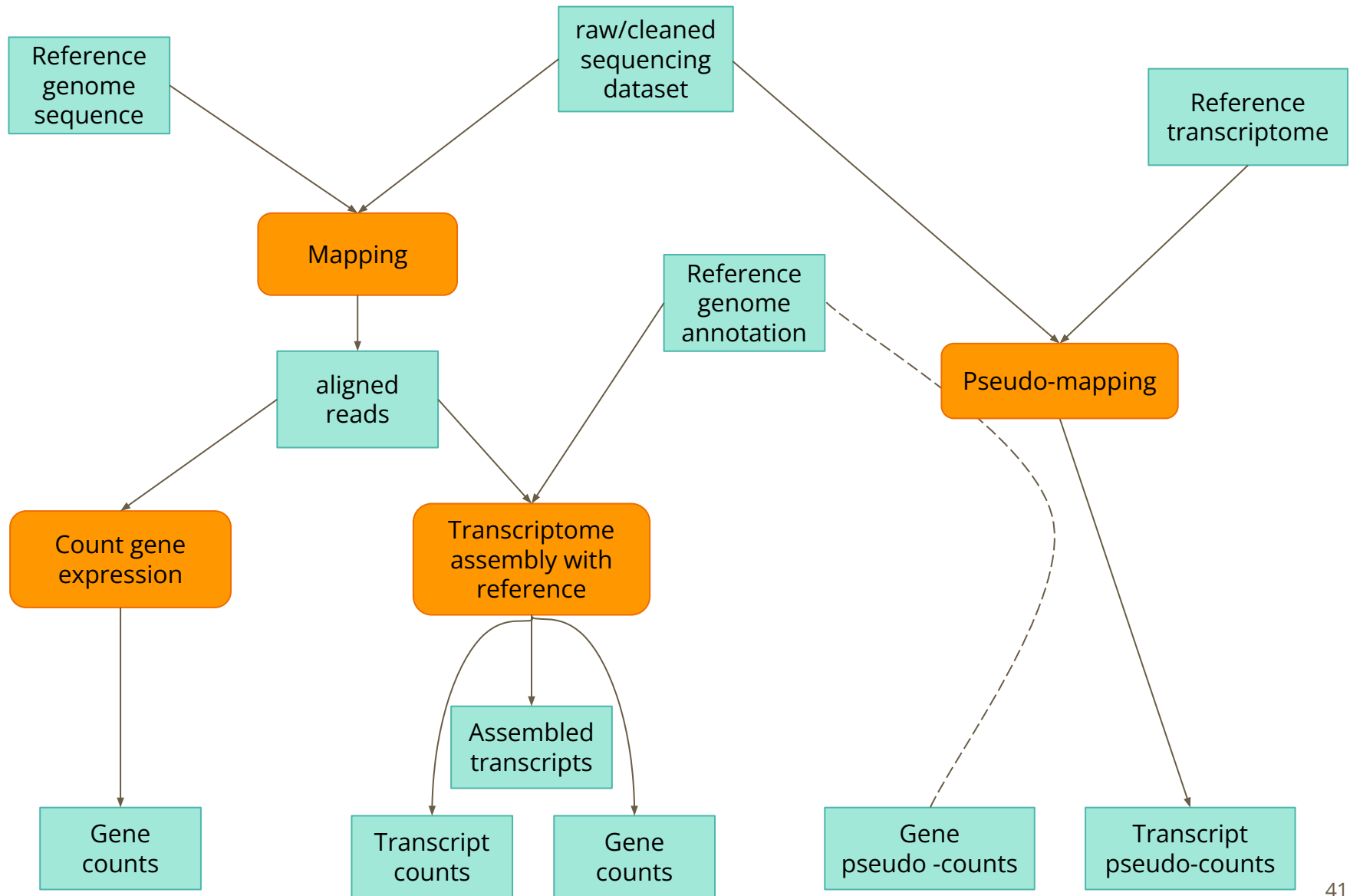
- Differential expression
  - between genes
  - between transcripts/isoformes
- Transcriptome assembly
  - variant calling
  - isoforme discovery

# What people do with their RNA-seq

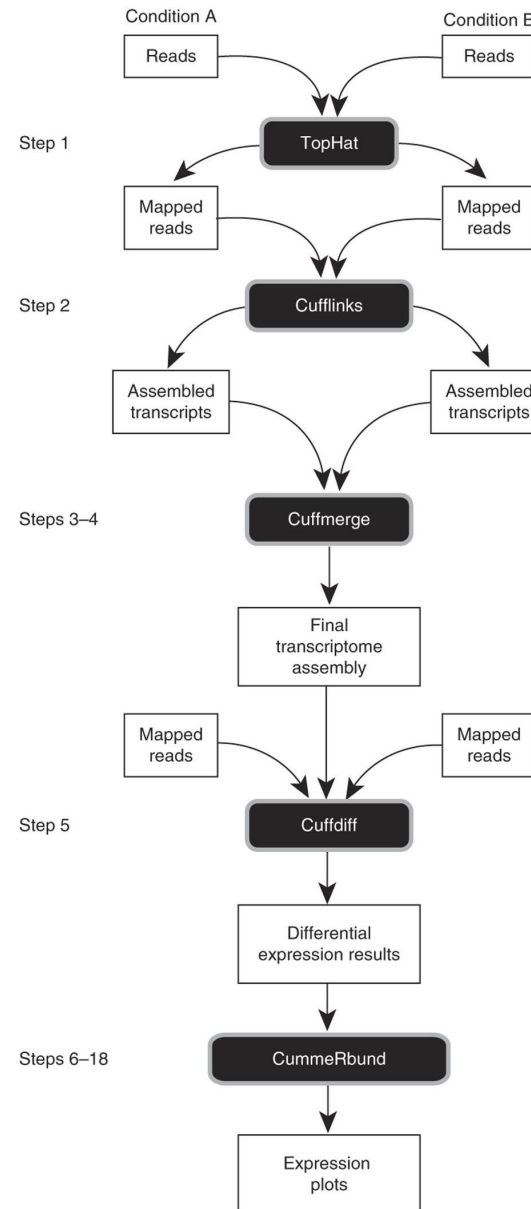
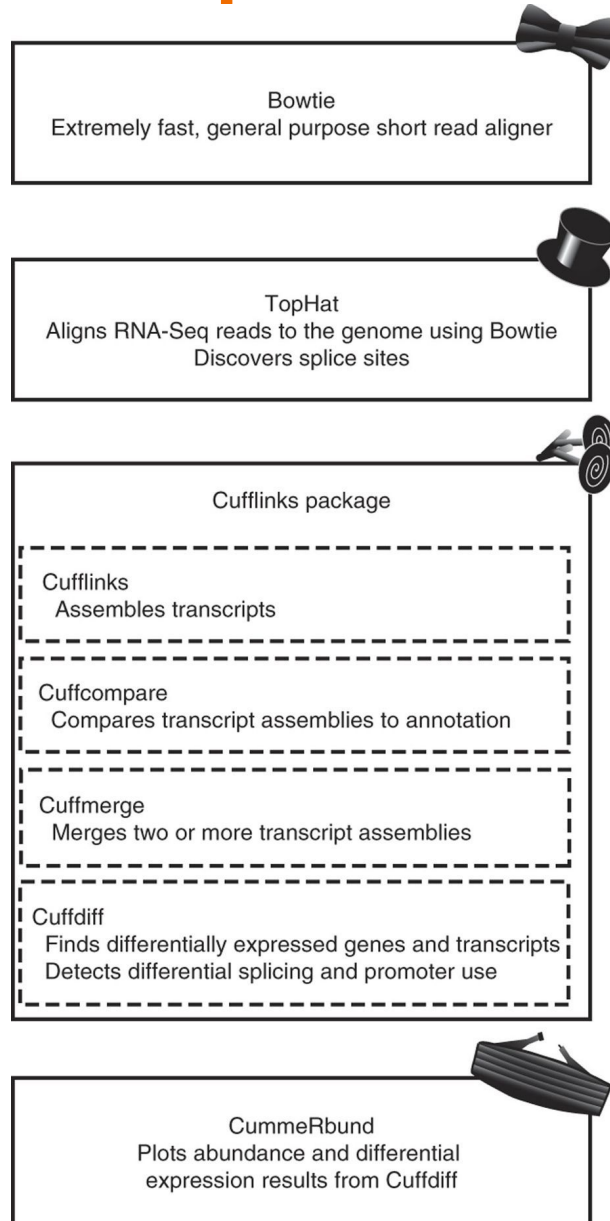




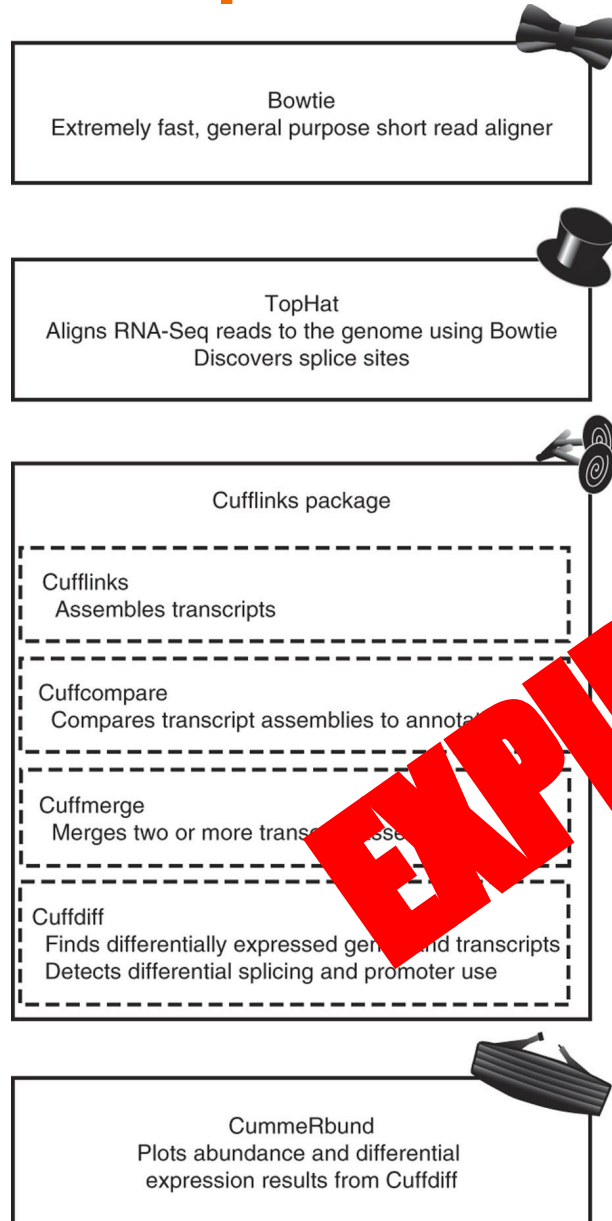
# RNA-seq w/ ref



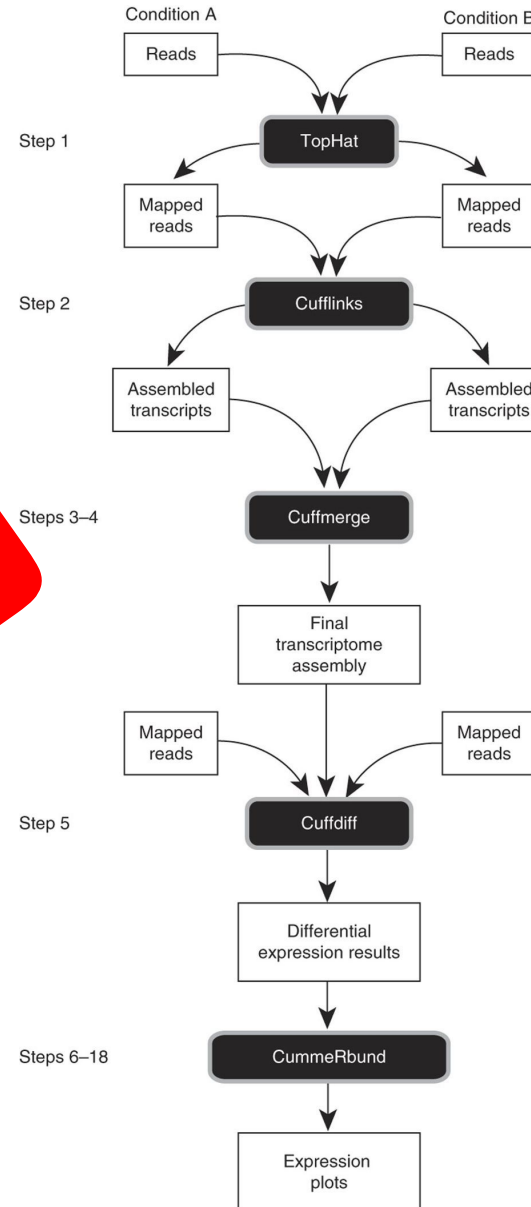
# The champion: Tuxedo Suite, "Classic" version



# The champion: Tuxedo Suite, "Classic" version



**EXPIRED**

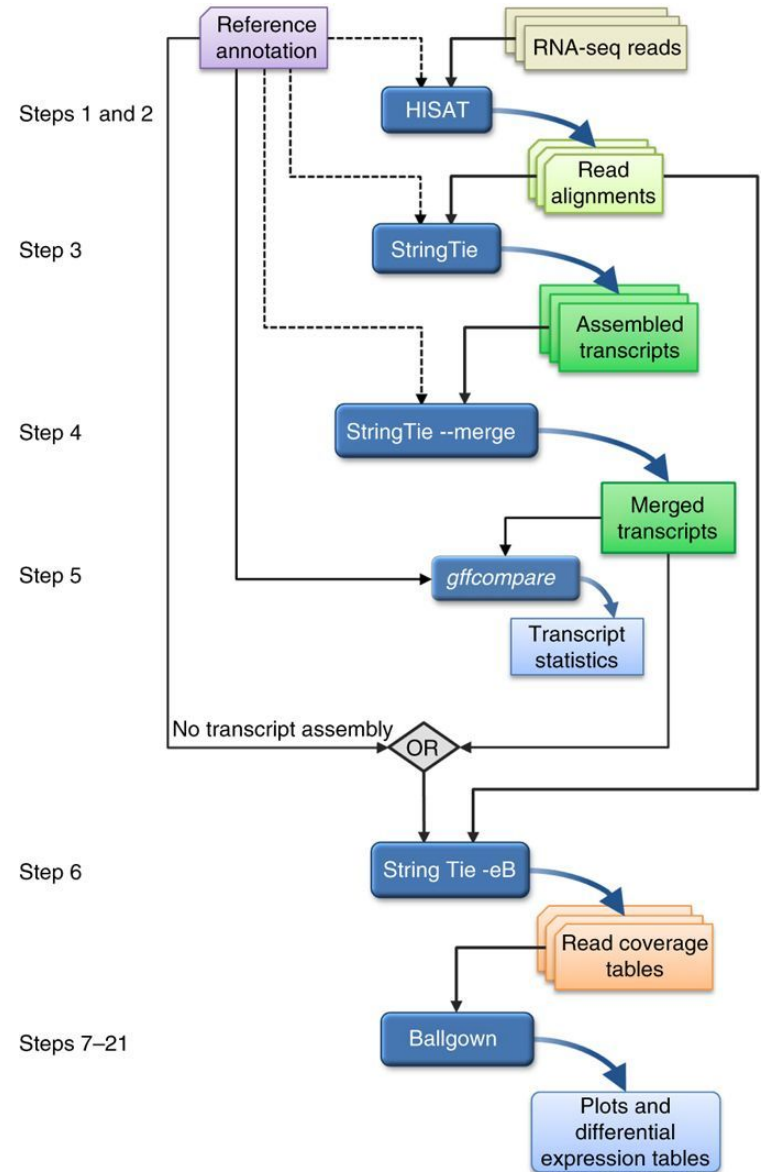


# The champion: Tuxedo Suite, New version

HISAT/HISAT2: splice aware aligner

StringTie: Transcriptome assembler

Ballgown: Differential expression analysis

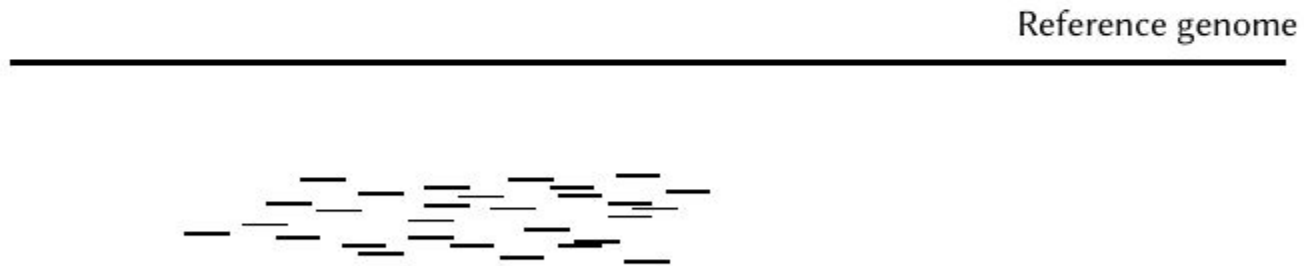


# Methods

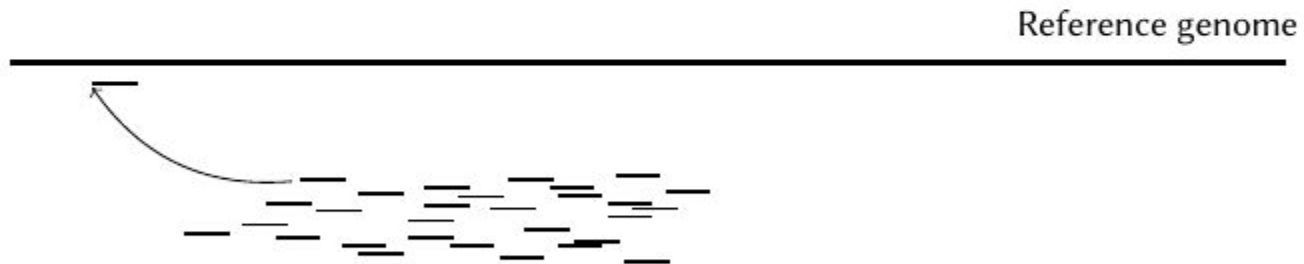
Let's focus on:

- 1 - read alignment/mapping/other techniques to assign a read to a gene or a transcript
- 2- strategies for counting gene expression from read assignation

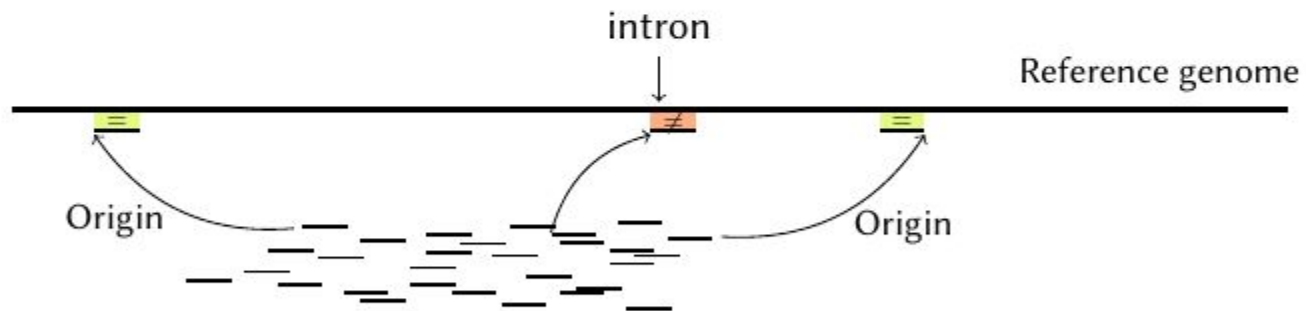
# RNA-Seq read mapping



# RNA-Seq read mapping

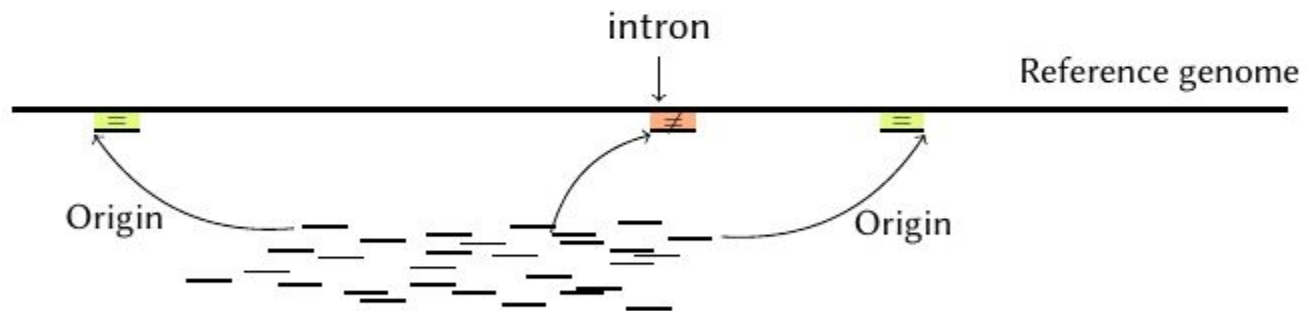


# RNA-Seq read mapping

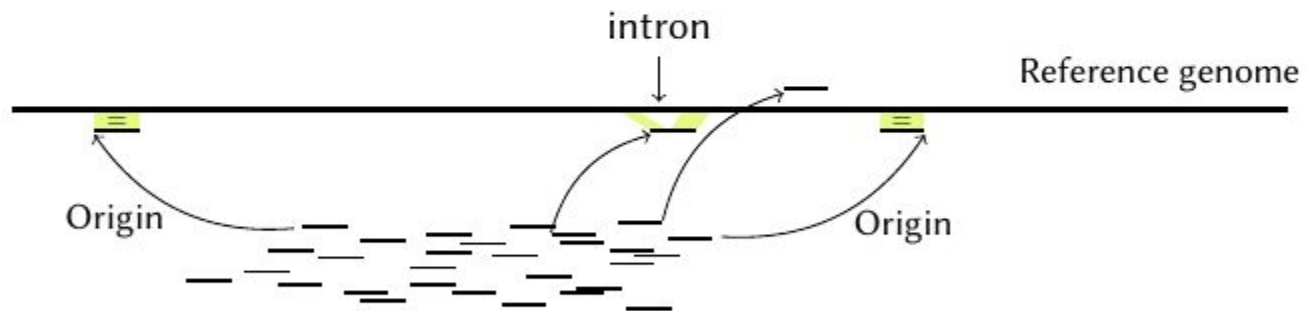




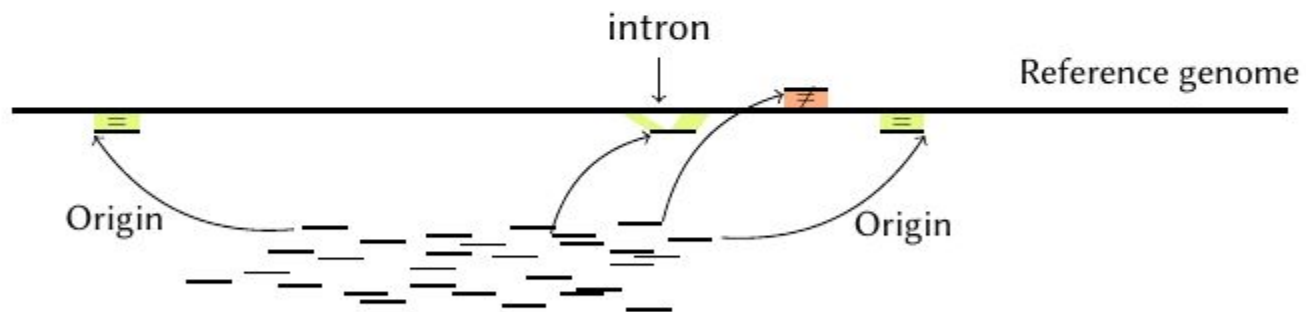
# RNA-Seq read mapping



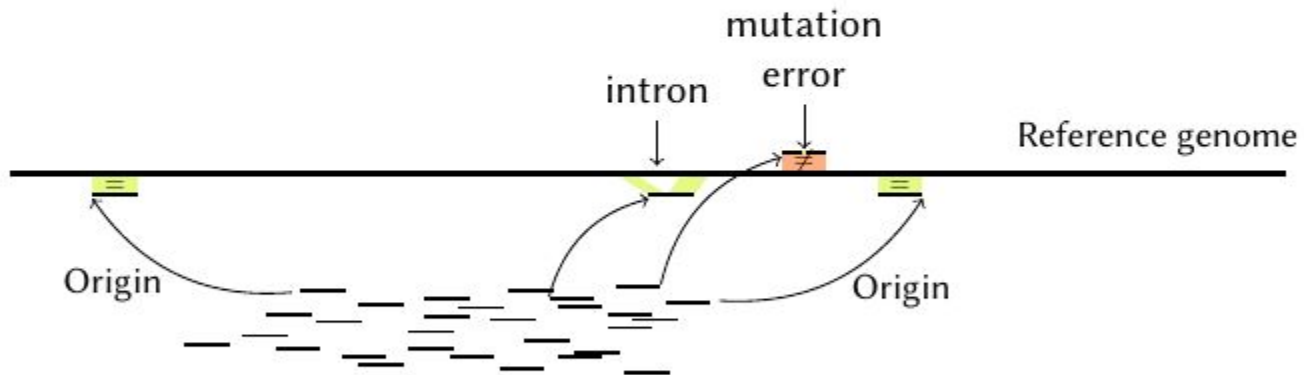
# RNA-Seq read mapping



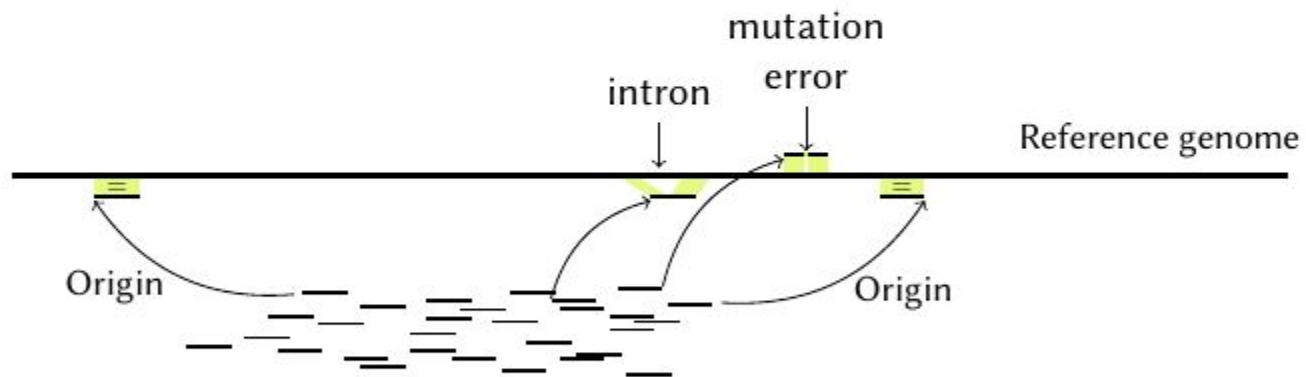
# RNA-Seq read mapping



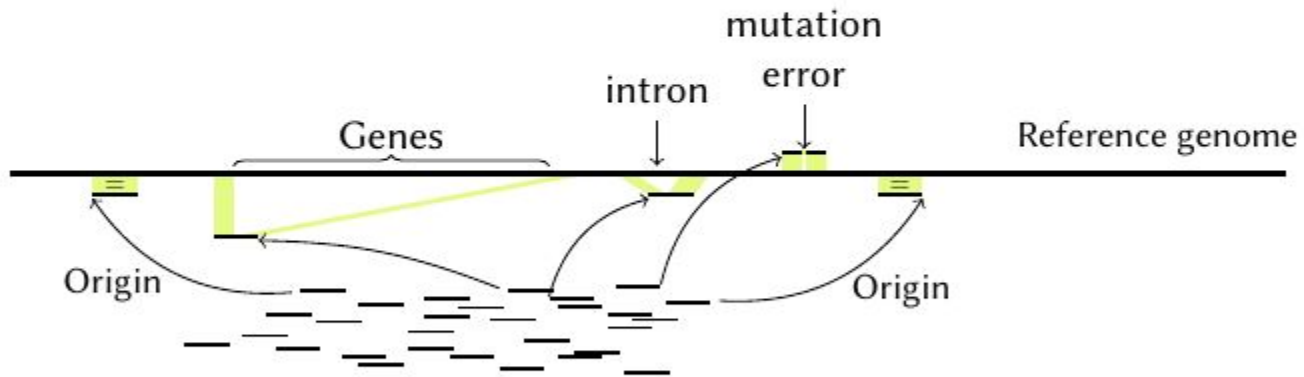
# RNA-Seq read mapping



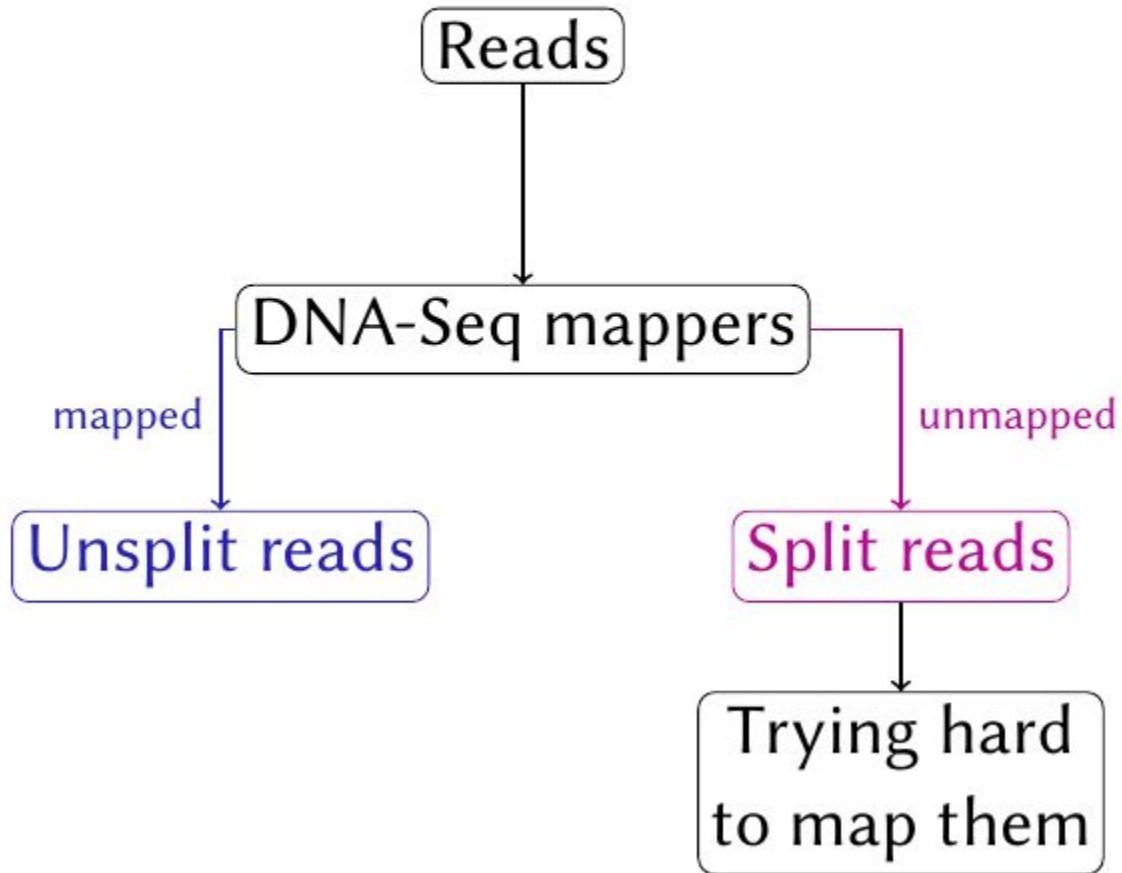
# RNA-Seq read mapping



# RNA-Seq read mapping



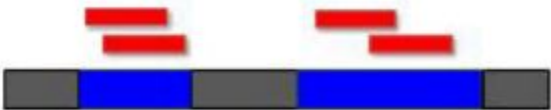
# Split reads don't align contiguously to the genome



# Mapping split reads by... splitting them – TopHat2

## (2) Genome alignment

Reads spanning a single exon are mapped

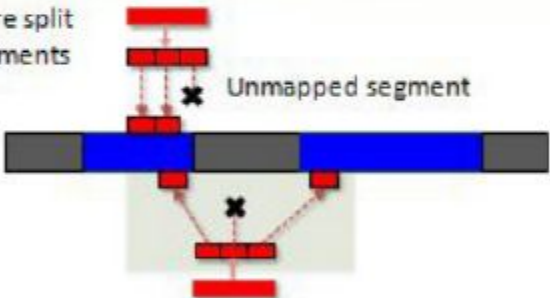


Multi-exon spanning reads are unmapped



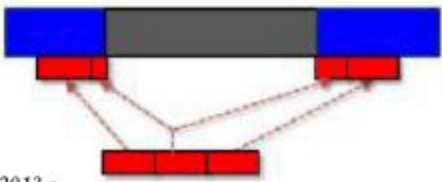
## (3) Spliced alignment

Reads are split into segments



(3-1) Segment alignment to genome

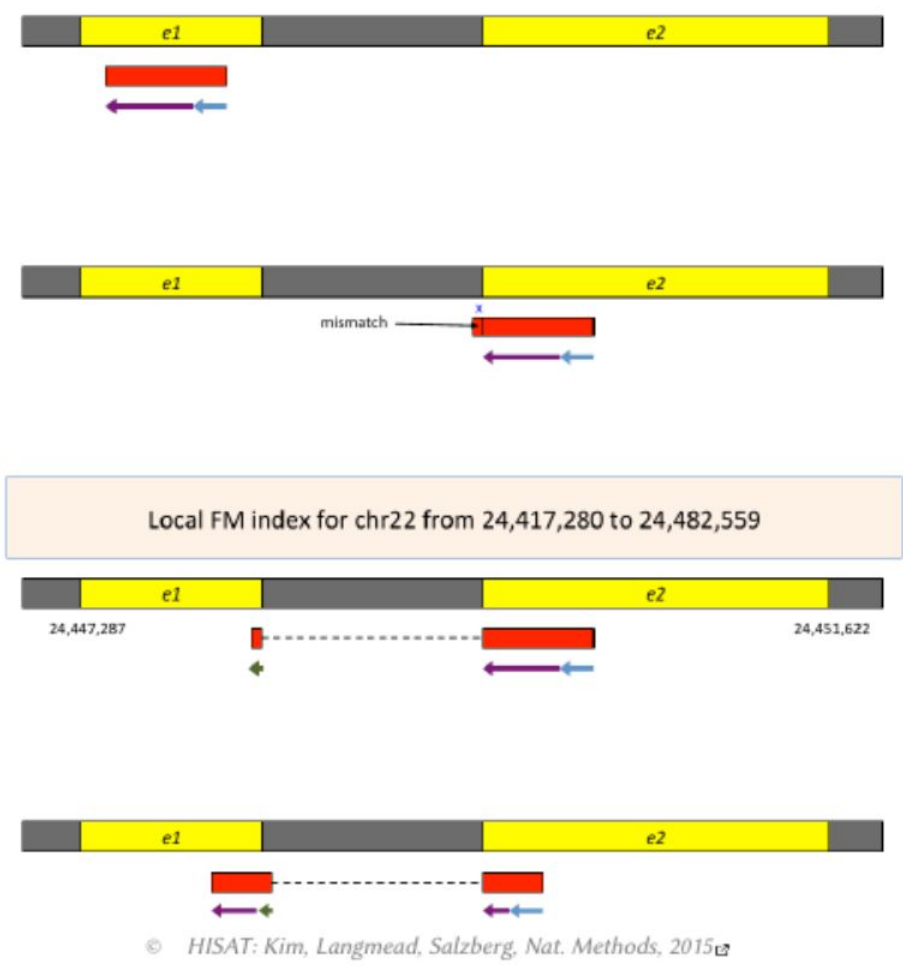
(3-2) Identification of splice sites (including indels and fusion break points)



© TopHat2: Kim et al, 2013



# Mapping all reads by splitting them – HISAT2, STAR



blablablaseul  
 dutexteblabla  
 ementblatexte

🔍 ctrl-f: seul

🔍 ctrl-f: seule

🔍 ctrl-f: ement

## Specificities of the approaches

### Mapping methods

<b>TopHat2</b>	Exact contiguous fixed-length
<b>HISAT</b>	Maximal mappable suffix
<b>STAR</b>	Maximal mappable prefix

### Indexing methods

<b>TopHat2</b>	FM-index
<b>HISAT</b>	Multiple FM-indices
<b>STAR</b>	Suffix Array

# Indexing methods

$$T = \overset{0}{C} \overset{1}{T} \overset{2}{A} \overset{3}{G} \overset{4}{T} \overset{5}{T} \overset{6}{A} \overset{7}{G} \overset{8}{\$}$$

TS	<b>8</b>	<b>6</b>	<b>2</b>	<b>0</b>	<b>7</b>	<b>3</b>	<b>5</b>	<b>1</b>	<b>4</b>
	\$	A	A	C	G	G	T	T	T
	C	G	G	T	\$	T	A	A	T
	T	\$	T	A	C	T	G	G	A
	A	C	T	G	T	A	\$	T	G
	G	T	A	T	A	G	C	T	\$
	T	A	G	T	G	\$	T	A	C
	T	G	\$	A	T	C	A	G	T
	A	T	C	G	T	T	G	\$	A
	<b>G</b>	<b>T</b>	<b>T</b>	<b>\$</b>	<b>A</b>	<b>A</b>	<b>T</b>	<b>C</b>	<b>G</b>

Burrows-Wheeler Transform

## *k*-mer sets - Burrows Wheeler transform?<sup>1</sup>

text            row\_row\_row\_your\_boat  
                 row\_row\_row\_your\_boat  
                 row\_row\_row\_your\_boat\$

**Burrows Wheeler transform (BWT)**

t r r r w w w w w w w o o o \_ \_ b b y y y r r r r r r r r r r r u u u t t t \$ \_ \_ \_ \_ \_ a a a o o o o o o o o o o o o o o \_ \_ \_

**Compression through run length encoding**

(t,1) (r,3) (w,9) (o,3) ... (\_,3)

---

<sup>1</sup>Adapted from Ben Langmead's course

## *k*-mer sets - Right contexts of w's

very similar right lexicographic contexts for all w's

```
row_row_row_your_boat  
row_row_row_your_boat  
row_row_row_your_boat$
```

ttrrrwwwwwwwwwwooo\_\_bbbyyrrrrrrrrrrruutt\$\_\_\_\_\_aaaooooooooooooo\_\_



## k-mer sets - Right contexts of o's

right lexicographic contexts for o's

row\_row\_row\_your\_boat  
row\_row\_row\_your\_boat  
row\_row\_row\_your\_boat\$

t r r r w w w w w w w w o o o \_ \_ b b b y y y r r r r r r r r r r r r u u u t t t \$ \_ \_ \_ a a a o o o o o o o o o o o o o o \_ \_ \_

# What approach is the best? (slide courtesy of J. Audoux)

NATURE METHODS | ANALYSIS

## Simulation-based comprehensive benchmarking of RNA-seq aligners

Giacomo Baruzzo, Katharina E Hayer, Eun Ji Kim, Barbara Di Camillo, Garret A FitzGerald & Gregory R Grant

METHOD | OPEN ACCESS

## A benchmark for RNA-seq quantification pipelines

Mingxiang Teng, Michael I. Love, Carrie A. Davis, Sarah Djebali, Alexander Dobin, Brenton R. Graveley, Sheng Li, Christopher E. Mason, Sara Olson, Dmitri Pervouchine, Cricket A. Sloan, Xintao Wei, Lijun Zhan and Rafael A. Irizarry

NATURE METHODS | ANALYSIS | OPEN

## Systematic evaluation of spliced alignment programs for RNA-seq data

Pär G Engström, Tamara Steijger, Botond Sipos, Gregory R Grant, RGASP Consortium, Gunnar Rättsch, Nick Goldman, Tim J Hubbard, Roderic Guigó & Paul Bertone

Article | OPEN

## Comparative assessment of methods for the fusion transcripts detection from RNA-Seq data

Am J Hum Genet. 2013 Oct 3; 93(4): 641-651.  
doi: [10.1016/j.ajhg.2013.08.008](https://doi.org/10.1016/j.ajhg.2013.08.008) PMID: PMC3

## Reliable Identification of Genomic Variants from RNA-Seq Data

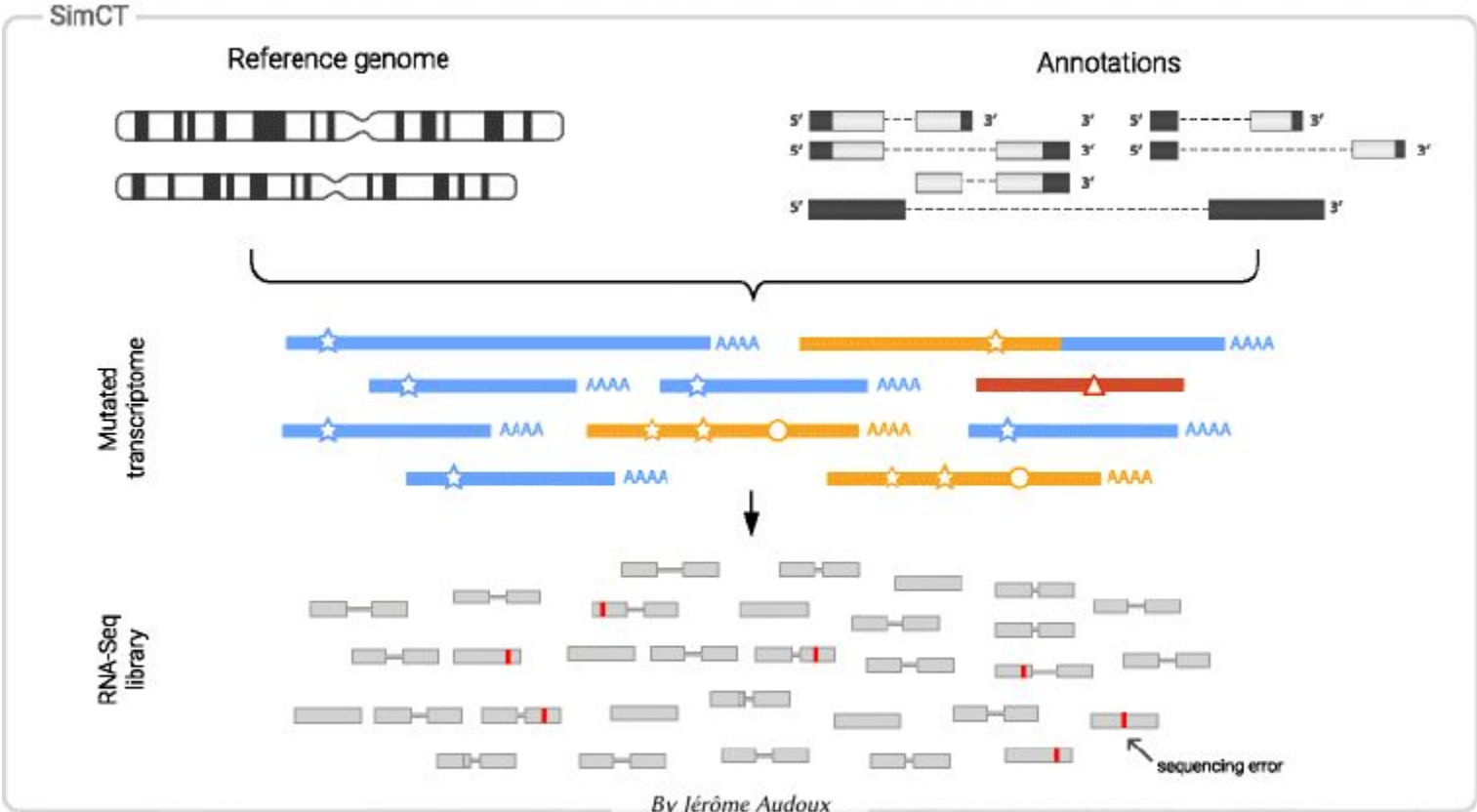
Robert Piskol,<sup>1</sup> Gokul Ramaswami,<sup>1</sup> and Jin Billy Li<sup>1,\*</sup>

Author information ► Article notes ► Copyright and License information ►

© Jérôme Audoux

# Benchmarking RNA-Seq aligners

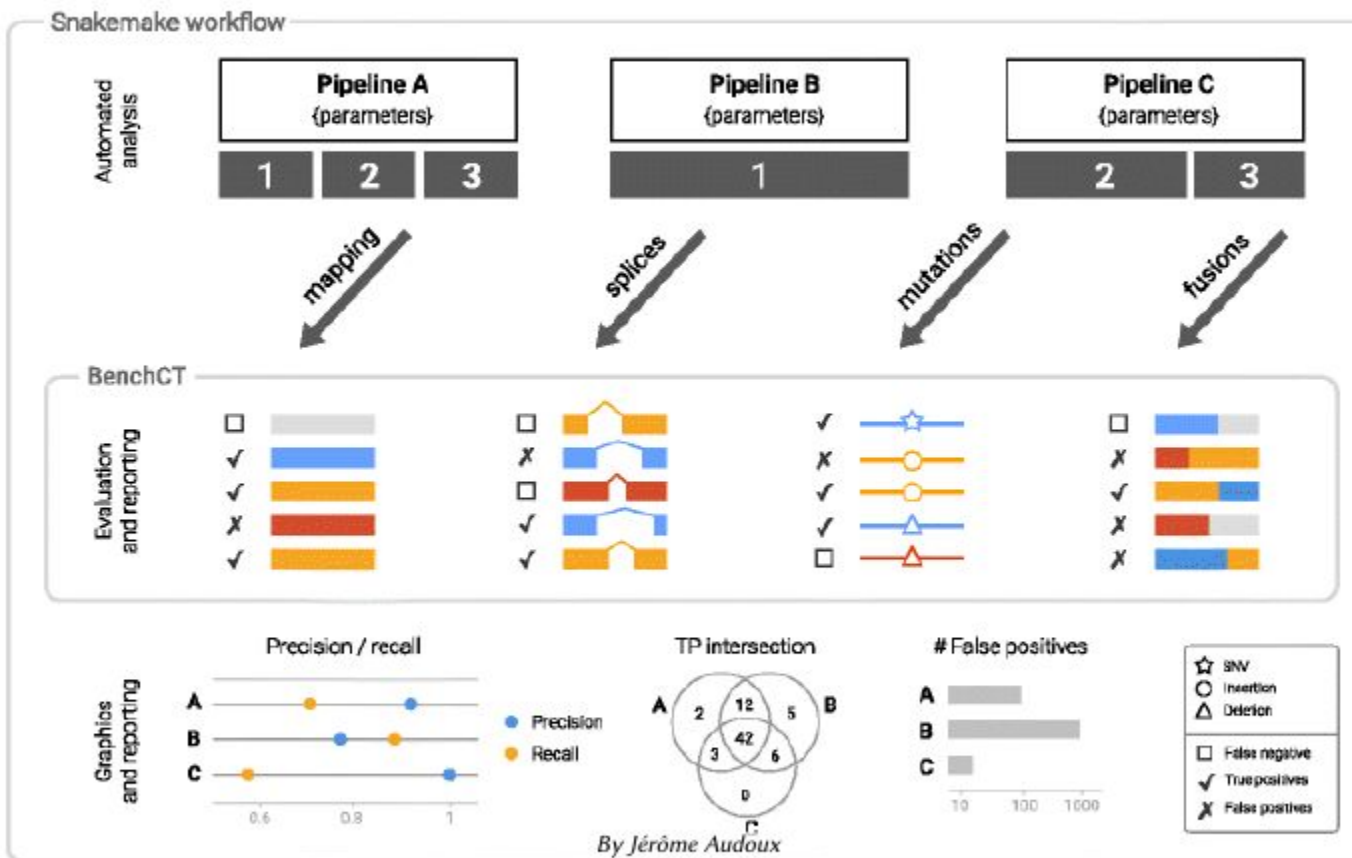
Audoux *et al*, BMC Bioinformatics, 2017





# Benchmarking RNA-Seq aligners

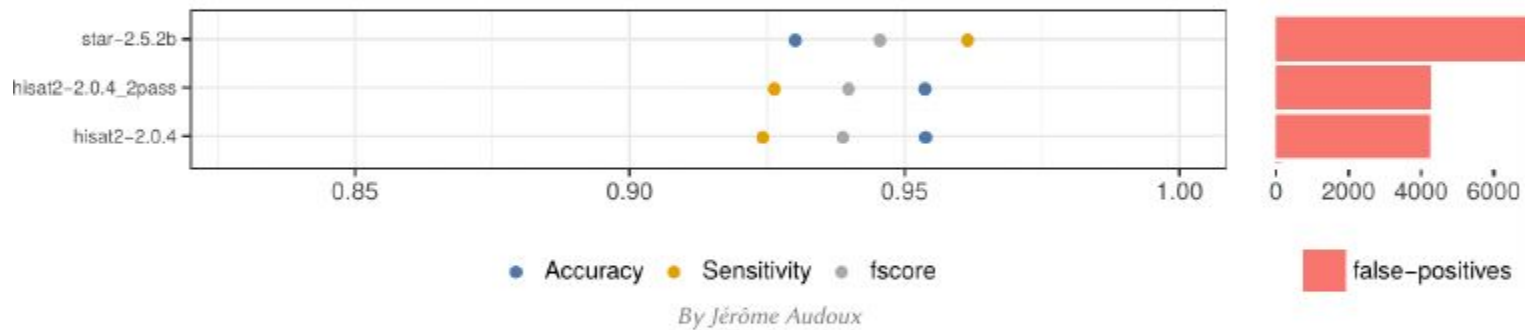
Audoux *et al*, BMC Bioinformatics, 2017 [↗](#)



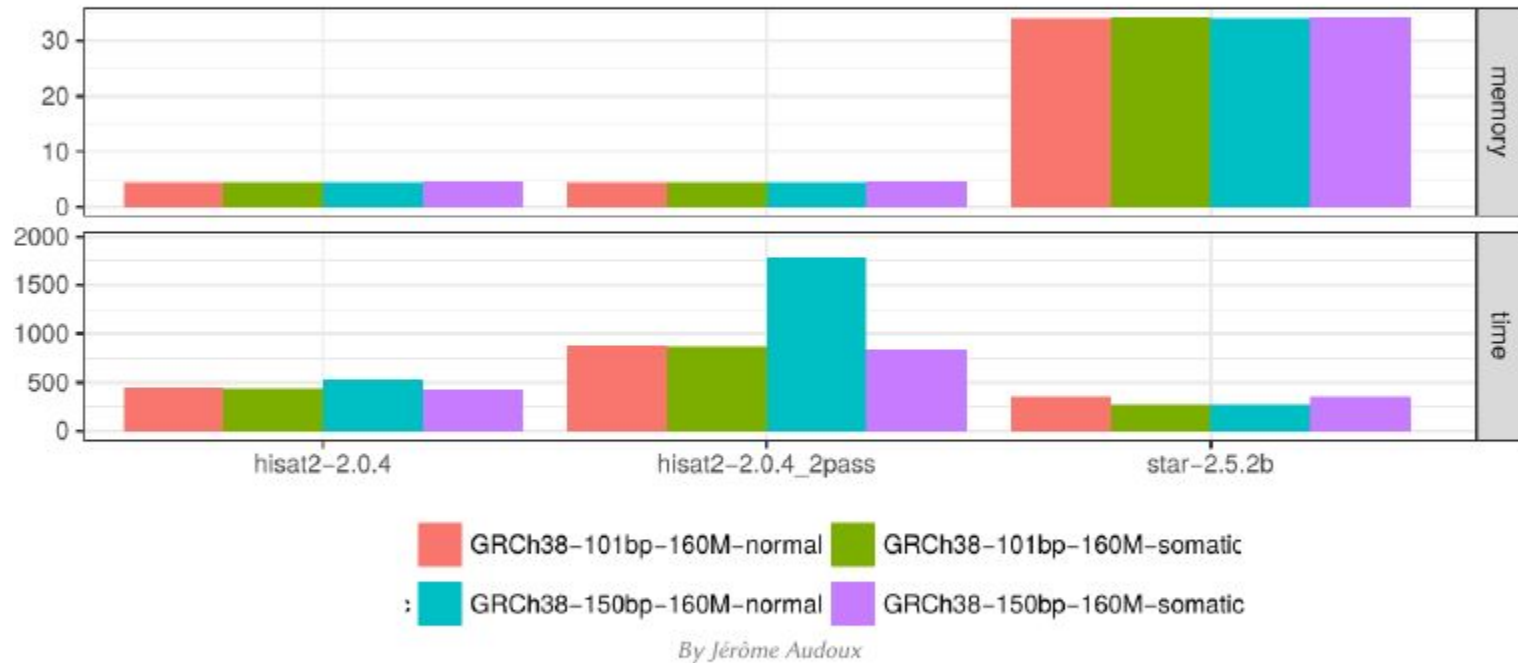
# STAR offers the best trade-off for splice detection

## Splicing

160M 150bp reads from GRCh38



# Space/time for read mappers



## Many people uses TopHat2

(> 10,623 citations in Scholar, > 1,000 citations in 2021 only)

Many people uses TopHat2

(> 10,623 citations in Scholar, > 1,000 citations in 2021 only)

**but don't**

**On TopHat2 website (since Feb 2016)** [↗](#)

TopHat2 « *is now largely superseded by HISAT2 which provides the same core functionality (i.e. spliced alignment of RNA-Seq reads), in a more accurate and **much more efficient** way* » .

Do you really need to map reads?

Does it matter to have a base pair precision location for hundreds of millions of reads?

## Alignment-free RNA-seq quantification

Quantifying transcripts may not require alignment

**Kallisto**

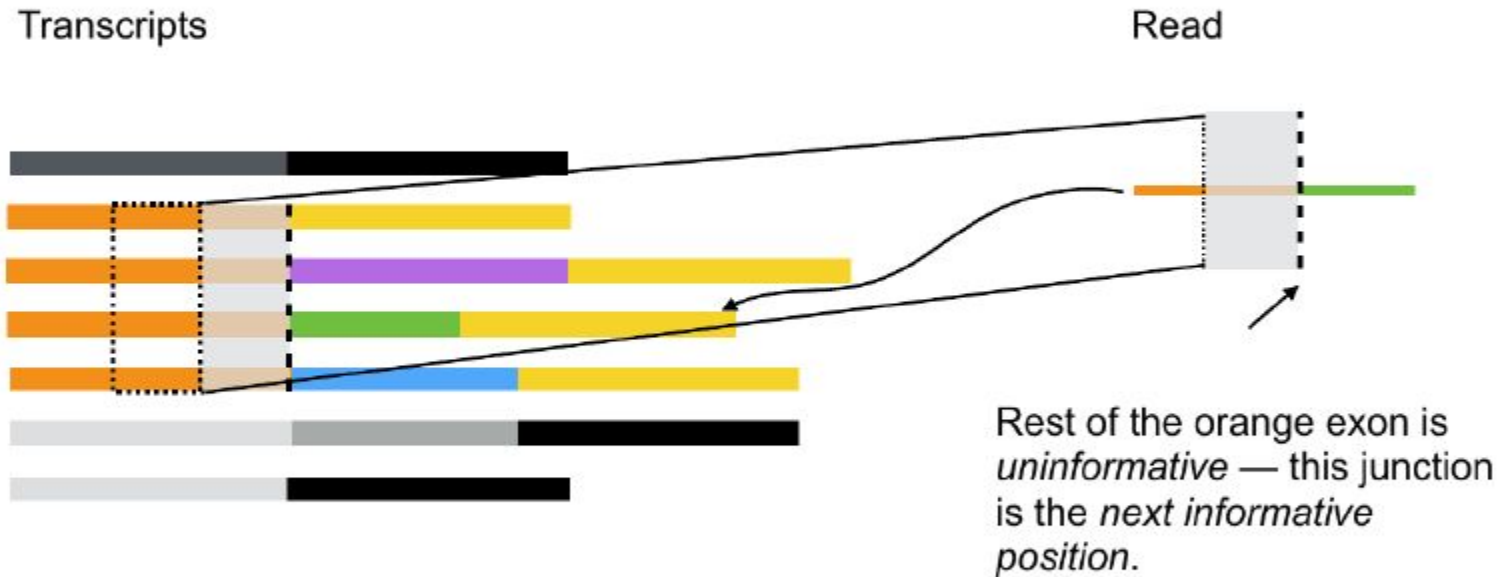
Bray et al, Nat. Biotechnology, 2016 [↗](#)

**Salmon**

Patro et al, Nat. Methods, 2017 [↗](#)

Two orders of magnitude faster than TopHat+Cufflinks

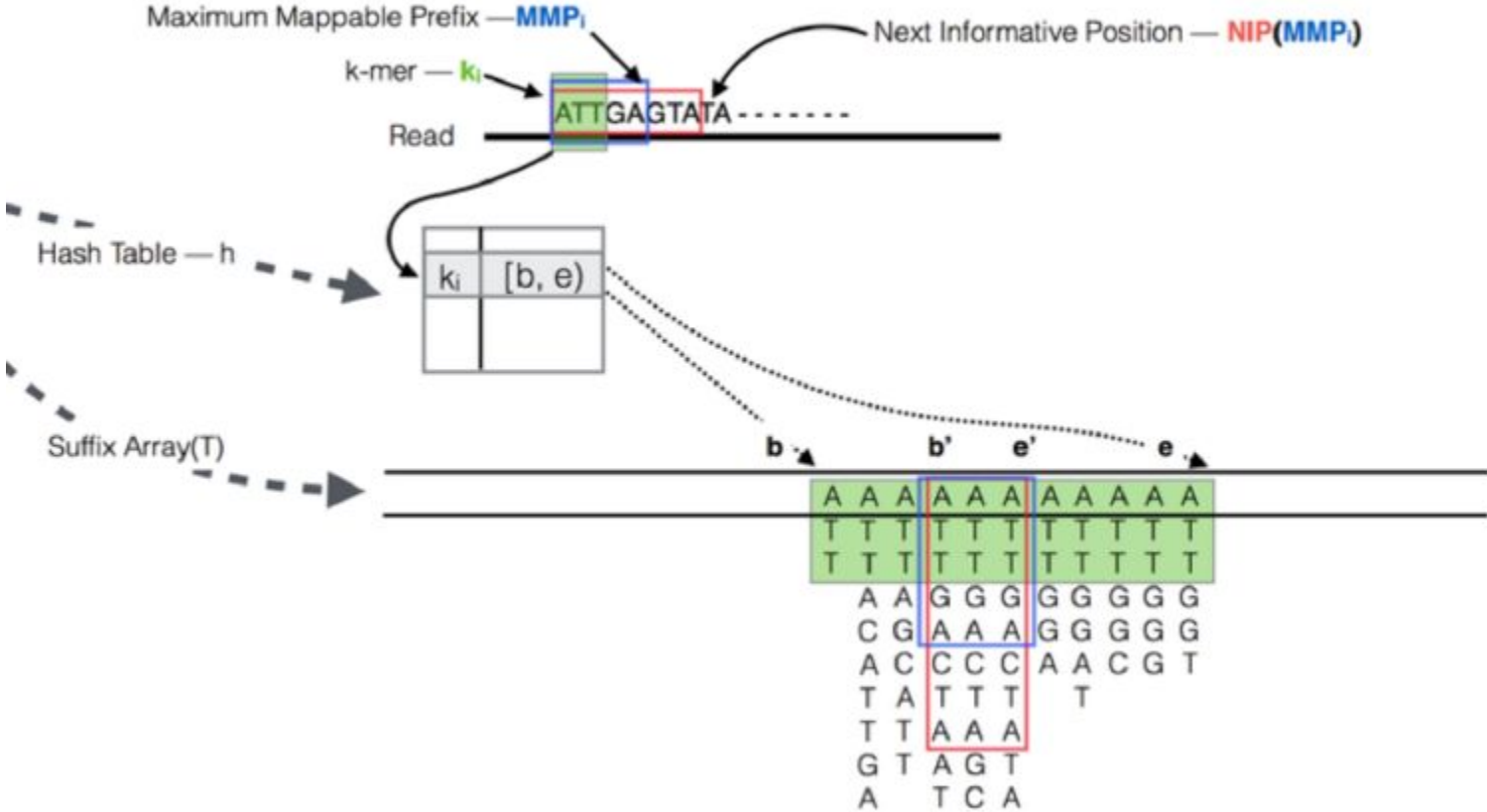
# How to quantify without aligning?



© Rob Patro (Salmon)



# How to quantify without aligning?



© Rob Patro (Salmon)

## Ultra fast methods with good results...

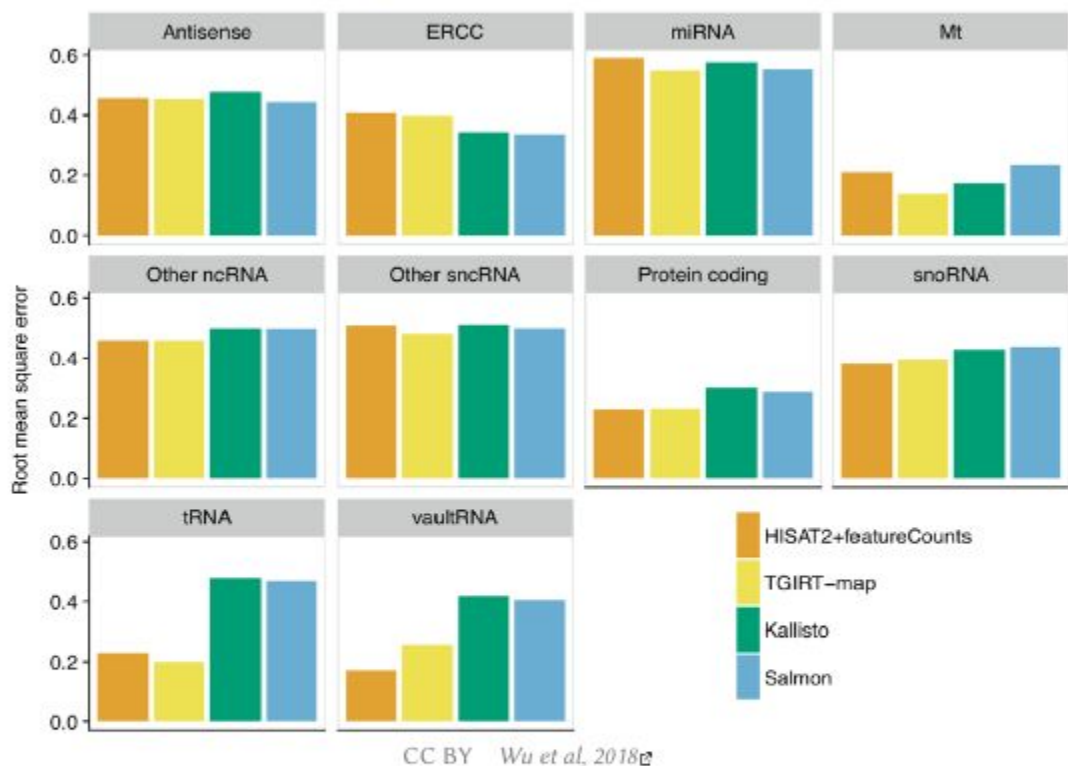
*« With the exception of the underperforming Flux Capacitor and eXpress, we found that the other algorithms performed similarly. »*

Teng et al, Genome Biology, 2016 [↗](#)

*« It is particularly noteworthy that Salmon, which (like Sailfish and Kallisto) bypasses traditional alignment and thereby quantifies a single sample in a matter of minutes, had a comparable performance to Cufflinks and RSEM. Importantly, we confirmed these results using a variety of assays on both empirical and simulated data. »*

Germain et al, Nucleic Acid Research, 2016 [↗](#)

## Good performances may not hold true for all the data



« We have found that alignment-based tools were more accurate in quantifying lowly-expressed or small genes. »

Wu et al, BMC Genomics, 2018

Where the differences come from?

1. Alignement vs pseudo-alignment
2. Genome reference vs transcriptome reference  
see Srivastava *et al*, 2020 [↗](#)

## Where the differences come from?

1. Alignement vs pseudo-alignment
2. Genome reference vs transcriptome reference  
see *Srivastava et al, 2020* [↗](#)
3. Quantification method

## How to quantify multi-mapped reads?

When a read maps at multiple loci,  
what transcript/gene should be counted?

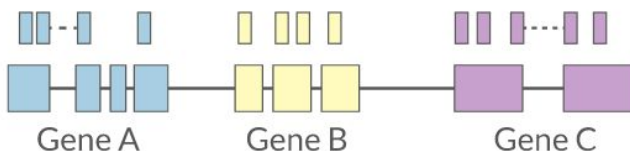
See Deschamps-Francoeur *et al*, 2020 [↗](#) (thanks Pierre!)

- ▶ None  
(*eg.* HTSeq-count, STAR genecount, featureCounts)
- ▶ Split counts evenly  
(*eg.* Cufflinks, featureCounts (with an option))
- ▶ Rescue based on single mapping reads  
(*eg.* Cufflinks (with an option))
- ▶ Expectation maximization  
(*eg.* RSEM, Salmon, Kallisto)

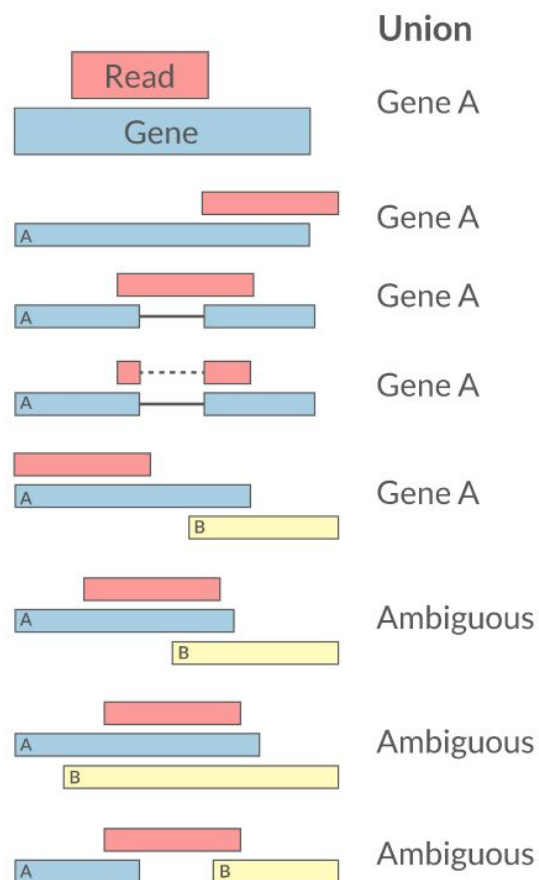
# Quantification - assignment and limits

## Quantification • Counts

- Read counts = gene expression
- Reads can be quantified on any feature (gene, transcript, exon etc)
- Intersection on gene models
- Gene/Transcript level

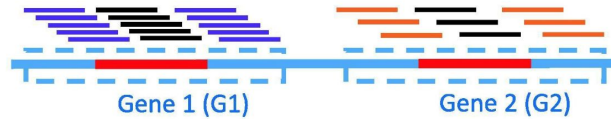


 featureCounts, HTSeq



# Counting gene expression from alignments

A



B

Approach to handle multireads	Read distribution representation	Counts
Ignore		G1: 10 reads G2: 6 reads
Count once per alignment		G1: 18 reads G2: 14 reads
Split them equally		G1: 14 reads G2: 10 reads
Rescue based on uniquely mapped reads		G1: 15 reads G2: 9 reads
Expectation-maximization		G1: 15 reads G2: 9 reads
Read coverage based methods		G1: 15 reads G2: 9 reads
Cluster methods		G1: 10 reads G2: 6 reads Cluster G1/G2: 8 reads

EM: counts are not bound to be integers!



# Counting gene expression from alignments

**Table 1**

Computational strategies and methods that handle multi-mapped reads.

Tool	Quantification level	Input	Strandedness can be specified	Count type	Strategy	Paired end	Confidence level	Focus
HTSeq-count	Gene	BAM	Y	Counts	Ignore	Y	N	Long RNA
STAR	Gene	Fastq	Y	Counts	Ignore	Y	N	Long RNA
geneCounts								
Cufflinks	Transcript	BAM	Y	RPKM	Split equally, Rescue	Y	N	Long RNA
featureCounts	Gene	BAM	Y	Counts	Ignore, count all, split equally	Y	N	Long RNA
CoCo	Gene	BAM	Y	Counts, CPM, TPM	Rescue	Y	N	Small RNA Long RNA
ERANGE	Transcript	BAM	N	RPKM	Rescue	Y	N	Long RNA
EMASE	Transcript	BAM	N	Counts, TPM	EM	Y	N	Long RNA
IsoEM2	Both	SAM	Y	FPKM, TPM	EM	Y	Confidence intervals	Long RNA
Kallisto	Transcript	Fastq	Y	TPM	EM	Y	Bootstrap values	Long RNA
RSEM	Both	Fastq, BAM	Y	Counts, TPM, FPKM	EM	Y	95% credibility intervals	Long RNA
Salmon	Transcript	Fastq	Y	Counts, TPM	EM	Y	Bootstrap values	Long RNA
MMR	N/A	BAM	Y	N/A	Read coverage	Y	N/A	Long RNA
MuMRRescueLite	Genomic loci	Custom format	N	Counts	Read coverage	N	N	Short sequence tags
Rcount	Gene	BAM	Y	Counts	Read coverage	N	N	Long RNA
ShortStack	Gene	Fastq, BAM	N	Counts, RPM	Read coverage	N	N	Small RNA
mmquant	Gene	BAM	Y	Counts	Gene Clustering	Y	N	Small RNA Long RNA
SeqCluster	Gene	BAM	N	Counts	Gene clustering	N	N	Small RNA
Fuzzy method	Gene	Custom format	N	Fuzzy counts	Fuzzy sets	N	Fuzzy counts	Small RNA Long RNA
geneQC	Gene	SAM	Y	NA	ML	Y	Mapping uncertainty level	Small RNA Long RNA

## How to quantify multi-mapped reads?

When a read maps at multiple loci,  
what transcript/gene should be counted?

See Deschamps-Francoeur *et al*, 2020 [↗](#) (thanks Pierre!)

- ▶ None  
(*eg.* HTSeq-count, STAR genecount, featureCounts)
- ▶ Split counts evenly  
(*eg.* Cufflinks, featureCounts (with an option))
- ▶ Rescue based on single mapping reads  
(*eg.* Cufflinks (with an option))
- ▶ Expectation maximization  
(*eg.* RSEM, Salmon, Kallisto)

# Methods still evolve: Salmon selective alignment

mapping



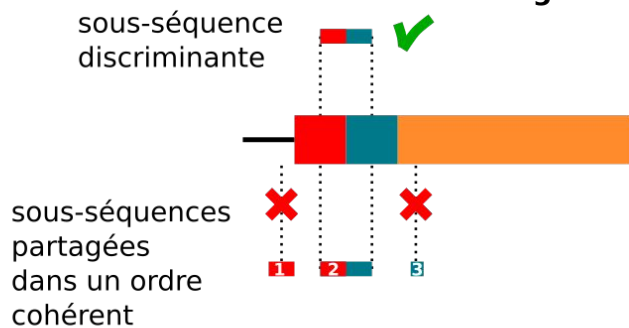
pseudo-mapping



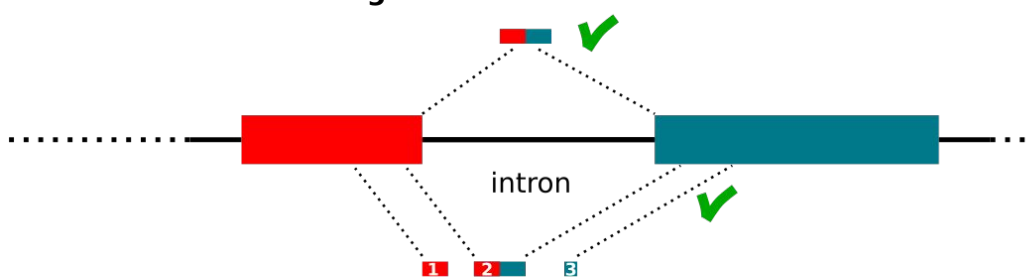
selective alignment



ailleurs sur le génome



notre gène



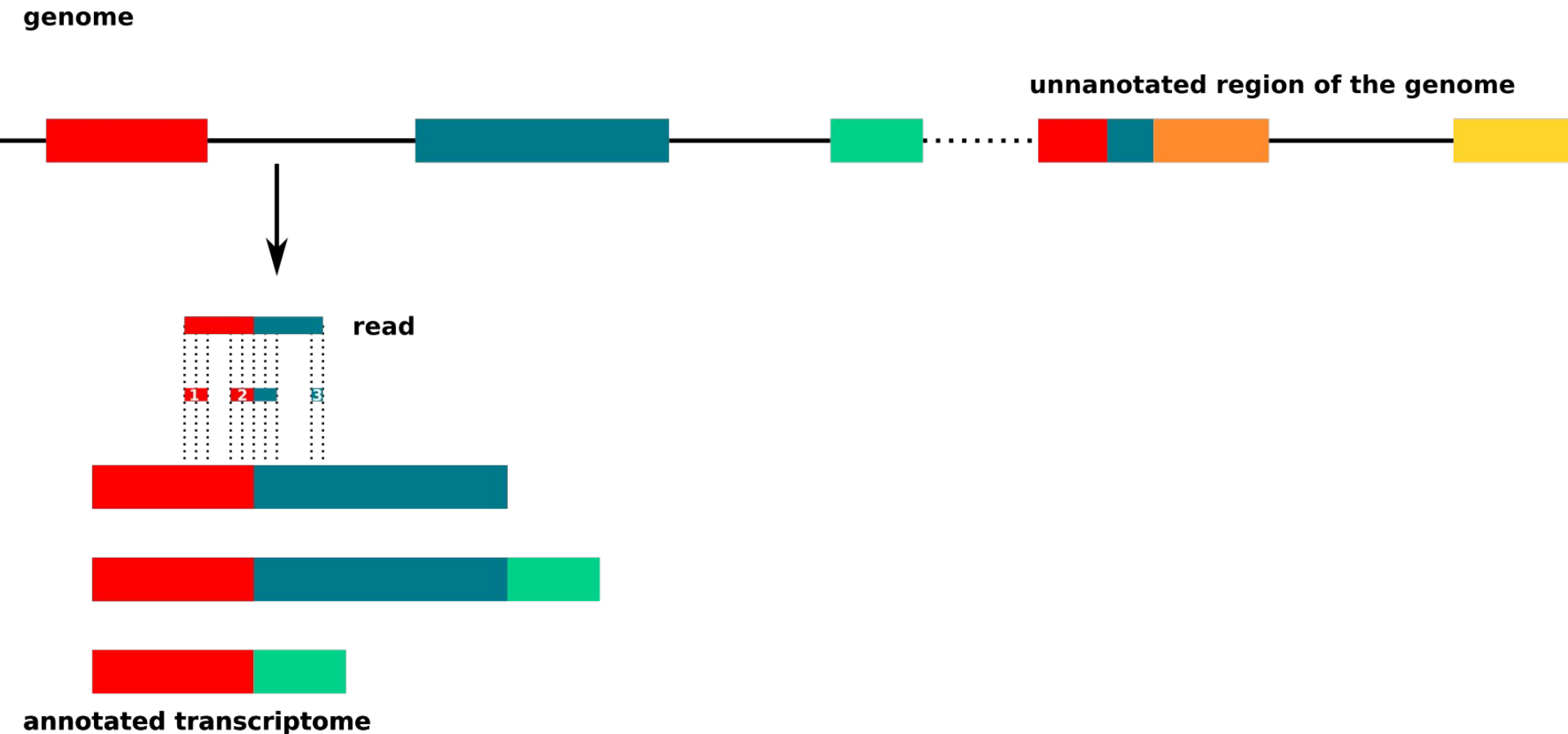
# Counting gene expression with Salmon

Salmon Selective alignment:

- inspired from long-read mapping techniques
- check that the **order** of shared subsequences is the same in the read and the targeted transcript (*chaining*)
- recent results\* suggest that this method **combines speed of mapping-free** approaches **and a quantification precision** closer to traditional mapping
- combined with a second idea...

\*Alignment and mapping methodology influence transcript abundance estimation, Srivastava et al. 2020

# Salmon: using genomic regions as decoys



# Salmon: using genomic regions as decoys

genome



unannotated region of the genome



read

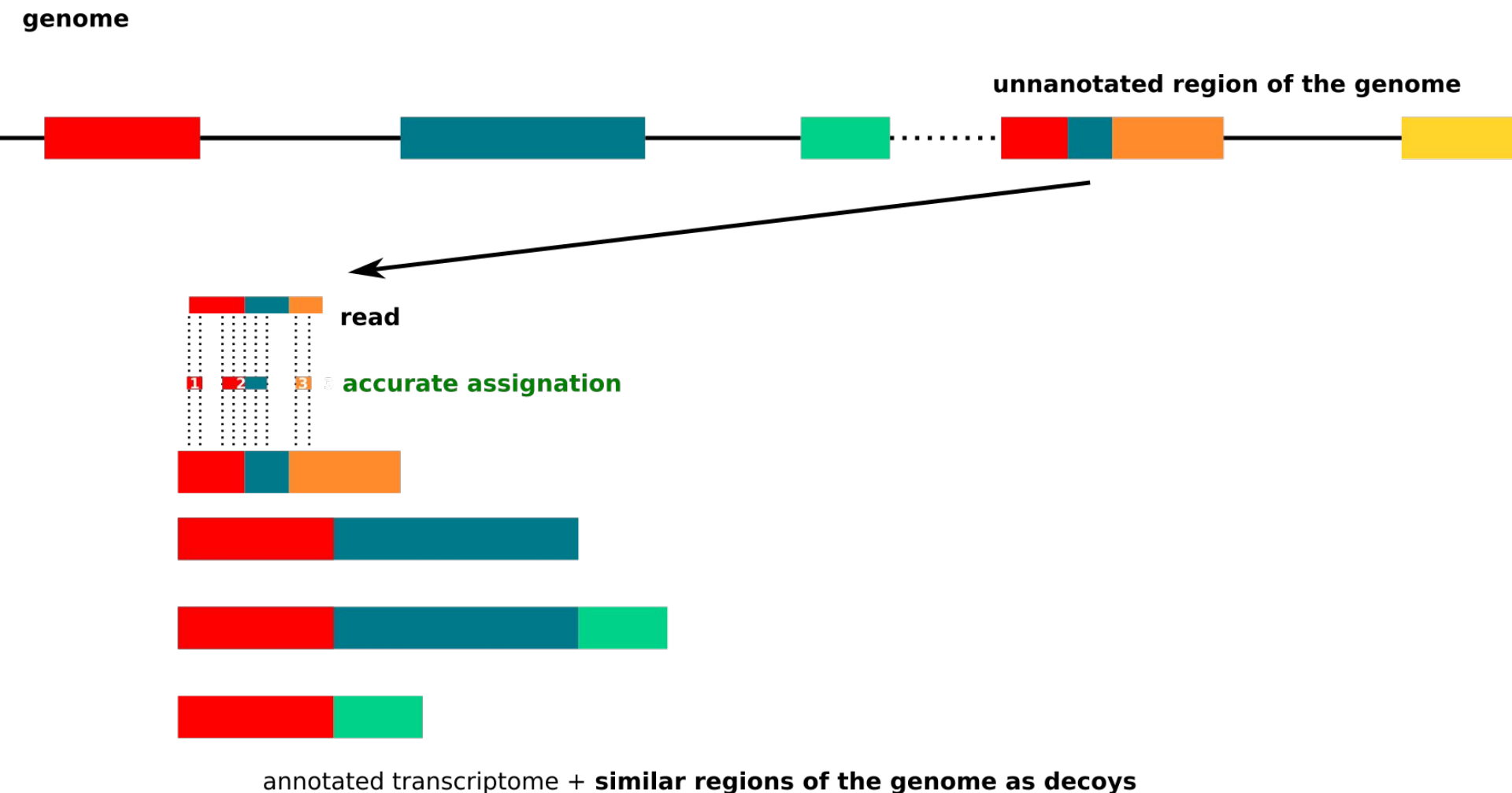


forced assignment

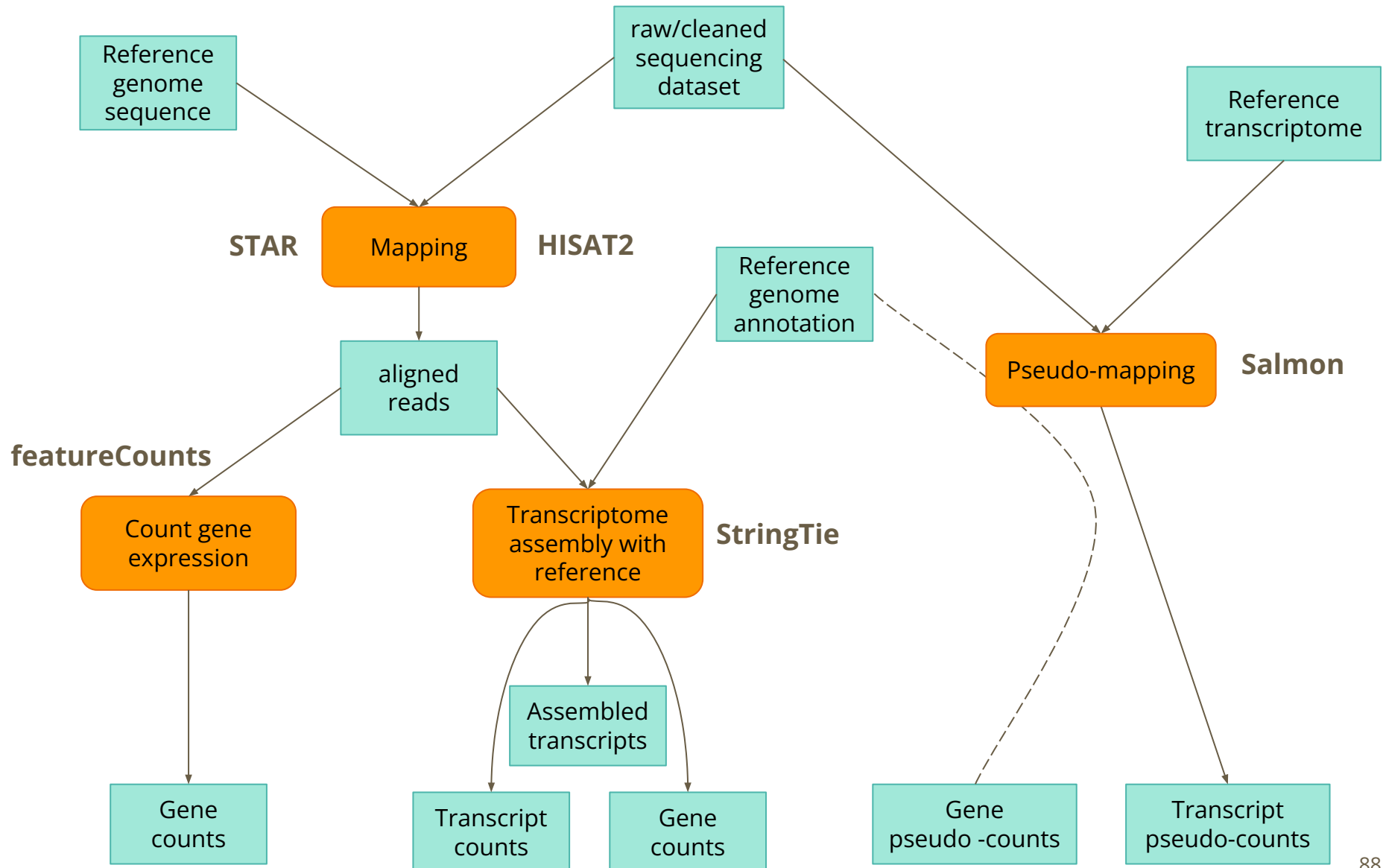


annotated transcriptome

# Salmon: using genomic regions as decoys



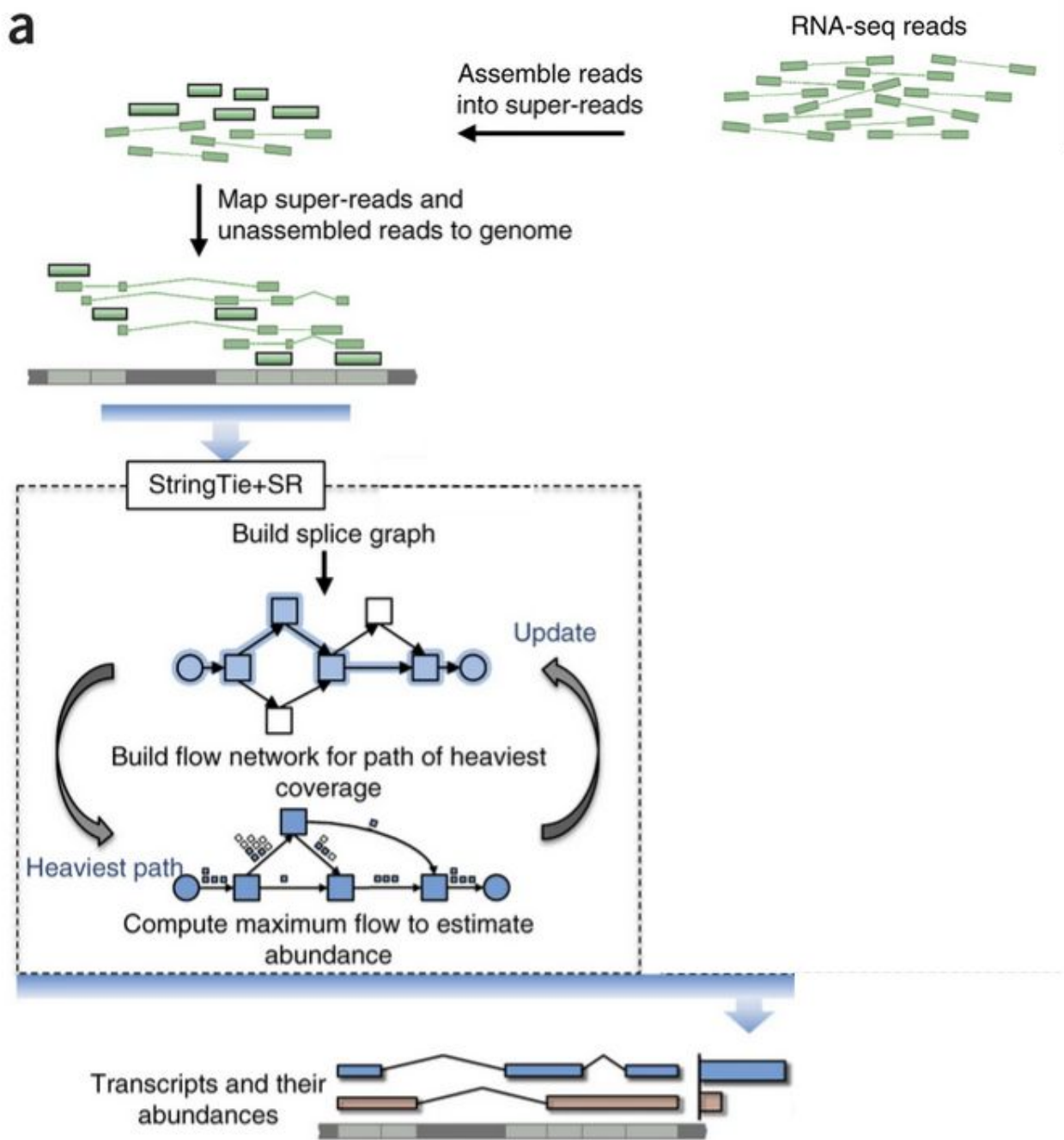
# RNA-seq w/ ref





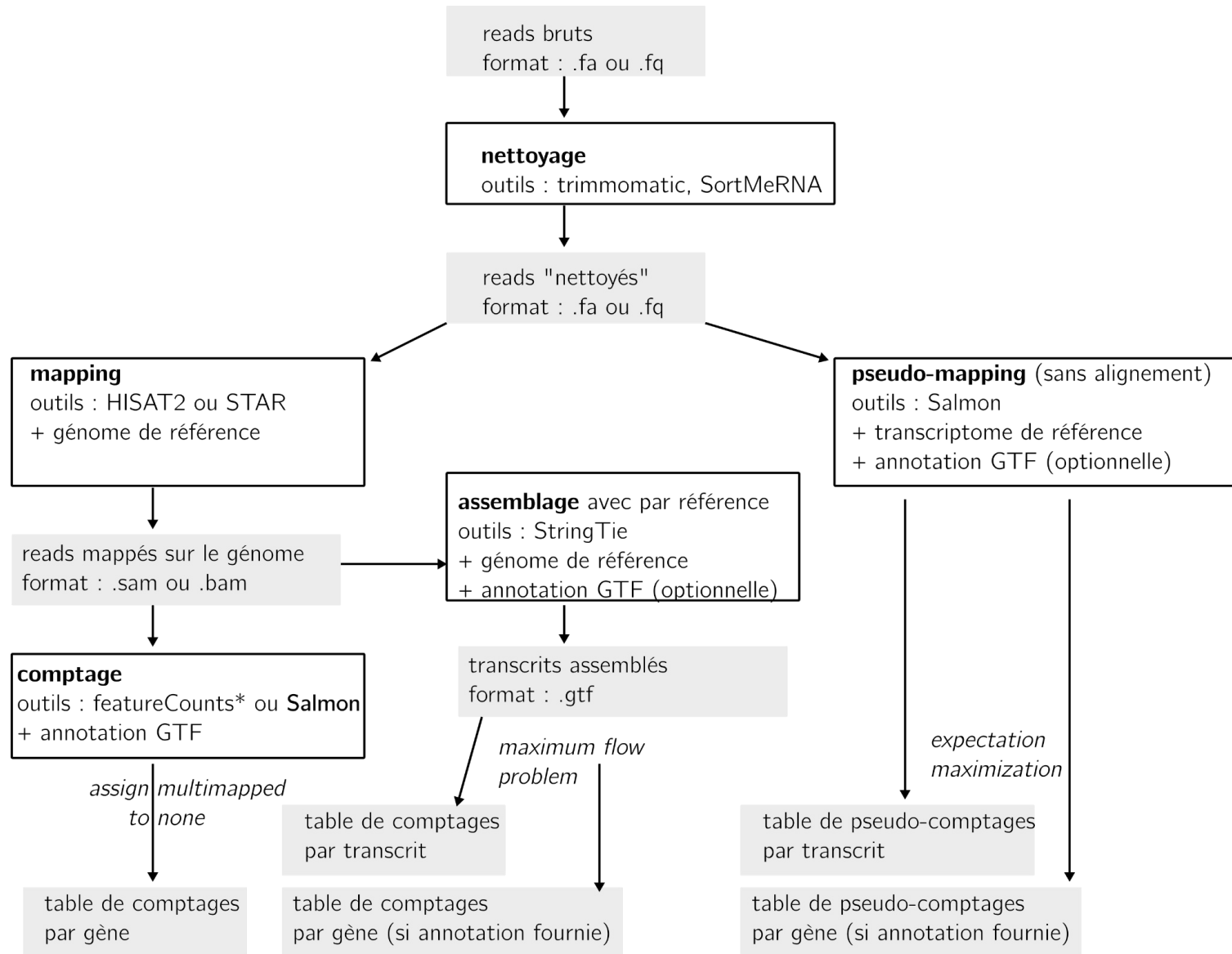
# Stringtie

adapted from  
Pertea et. 15



# Practical: Mapping and Quantification

# Working with a reference: our achievements



\* à vocation pédagogique pour le TP, déconseillé par ailleurs.

# Recommended pipeline (as of Oct 2023)

- Transcript/Gene quantification with mapping: STAR + Salmon
- Mapping-less transcript quantification: Salmon or Kallisto
- Transcriptome assembly: HISAT2 + StringTie (+ Ballgown ?)

# *De novo* RNA-seq

# *De novo* approaches

- ❑ *De novo* methods are approaches that are **free from a reference** for producing results
- ❑ Reference-based approaches have limitations as **results depends on the quality of the reference**
- ❑ Sometimes we don't even have a reference
- ❑ *De novo* and reference-based are **complementary**

# Why do we need *de novo* approaches

Aren't references good enough?

- ❑ Disease-associated transcripts
- ❑ Genetic polymorphism in transcripts
- ❑ *de novo* methods are helping creating tomorrow's references

Abstract

Reference transcriptomes:  
the making of



Enter direct RNA-seq  
assembly

Shall we ever reach a  
complete reference  
transcriptome?

Ignore non-reference  
transcripts at your own  
risks

Opinion | Open Access

## Bridging the gap between reference and real transcriptomes

[Antonin Morillon](#) and [Daniel Gautheret](#)  

*Genome Biology* 2019 20:112

<https://doi.org/10.1186/s13059-019-1710-7> | © The Author(s). 2019

Published: 3 June 2019

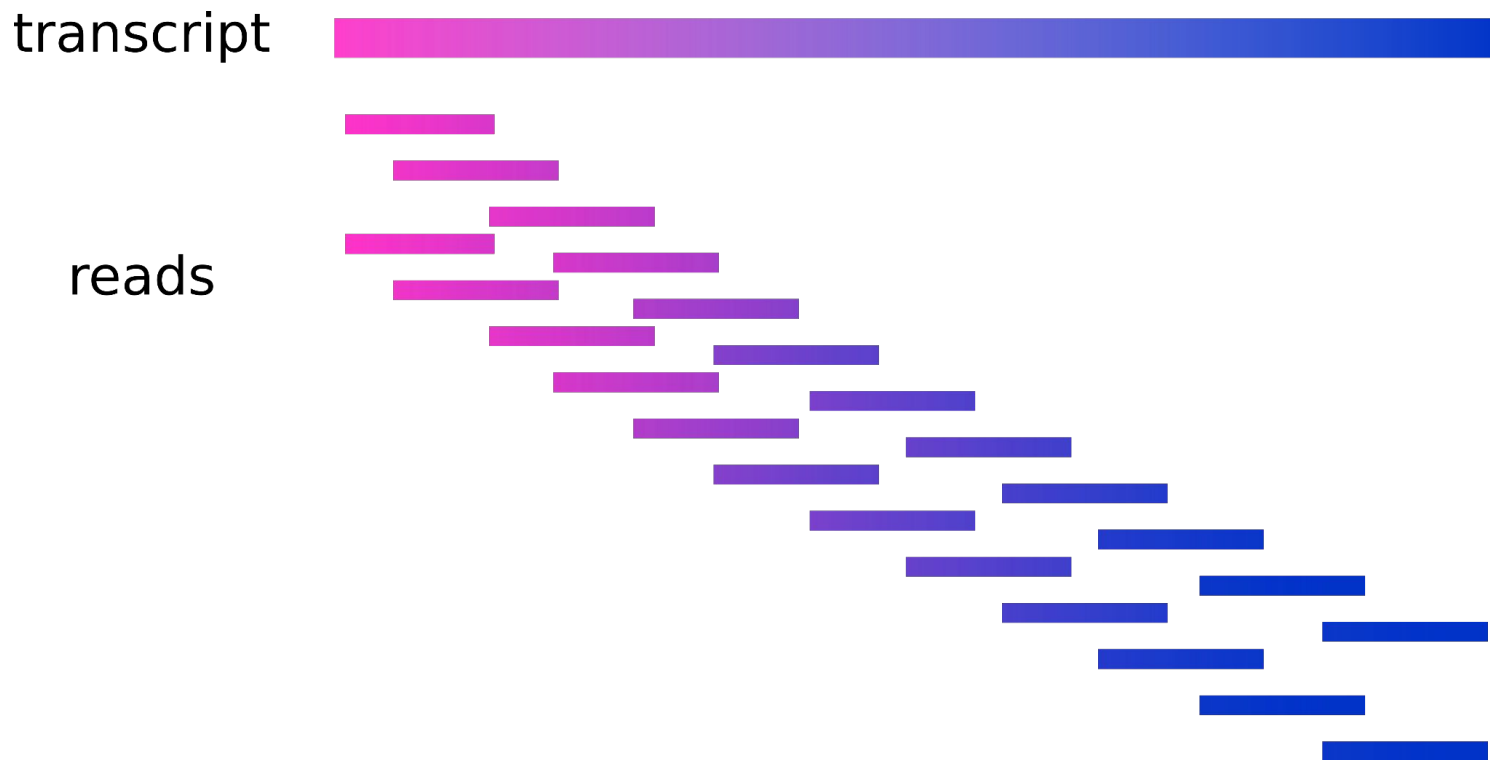
The more novel and specific is your need, the more likely you need new bioinformatics (and *de novo*)



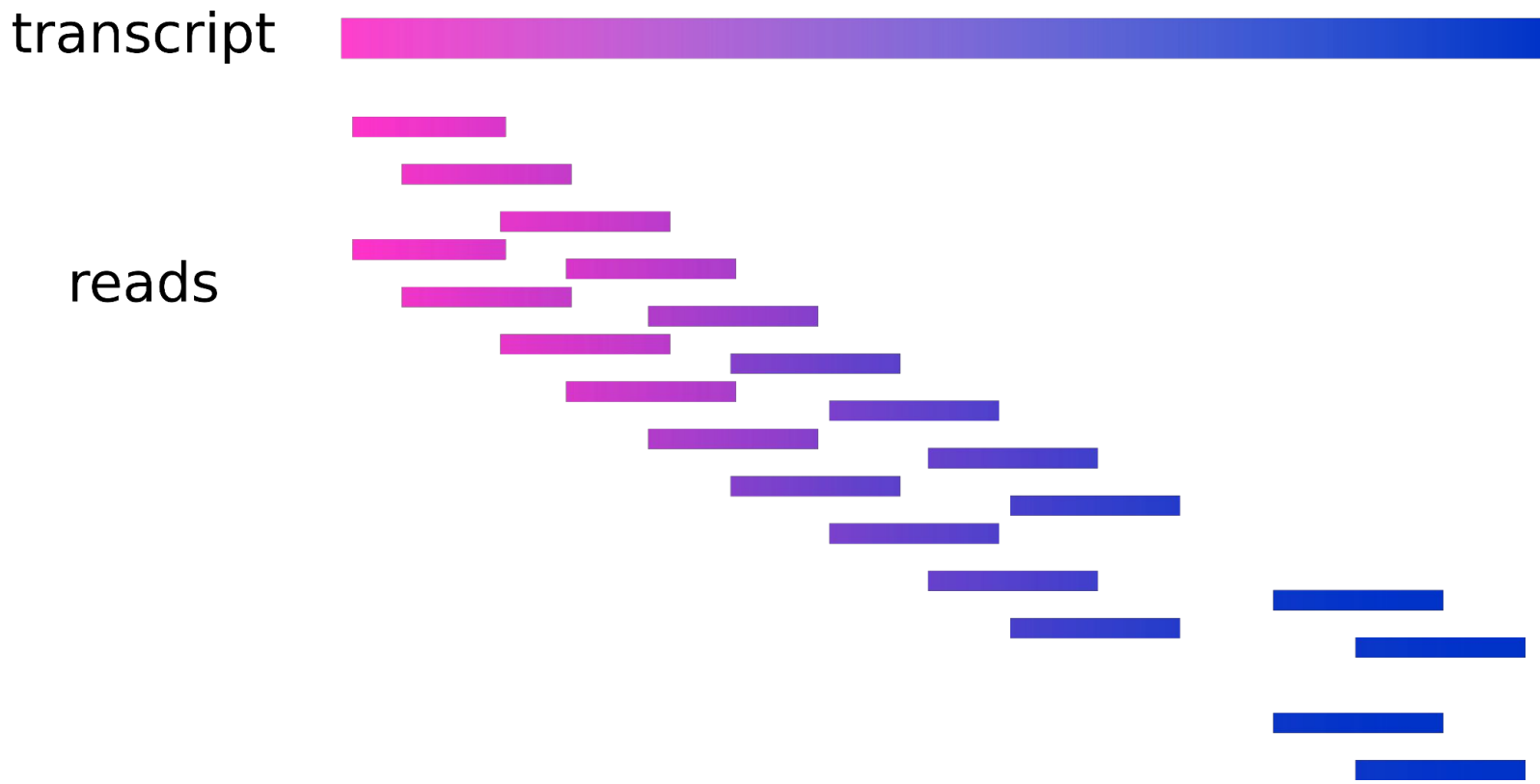
# What can be done with *de novo* methods

- ❑ transcript assembly + quantification
- ❑ genetic polymorphism detection
- ❑ alternative transcript detection + quantification

# The *de novo* assembly challenge



# The *de novo* assembly challenge



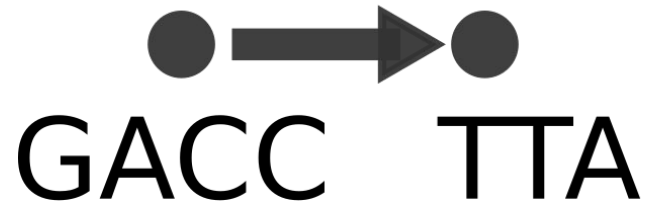
# The *de novo* assembly challenge



# Assembly recap

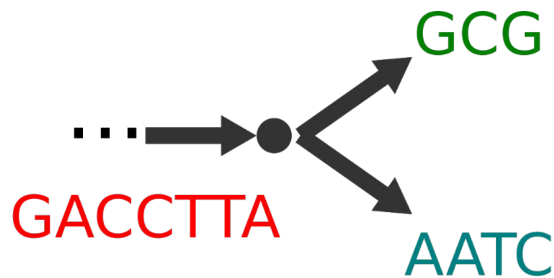
Assembly is like taking a step after another in a maze

One step is a group of nucleotides



# Assembly recap

Until you have a choice to make :



why does this happen? check the reads:

CTTAGCG

TTAAATC

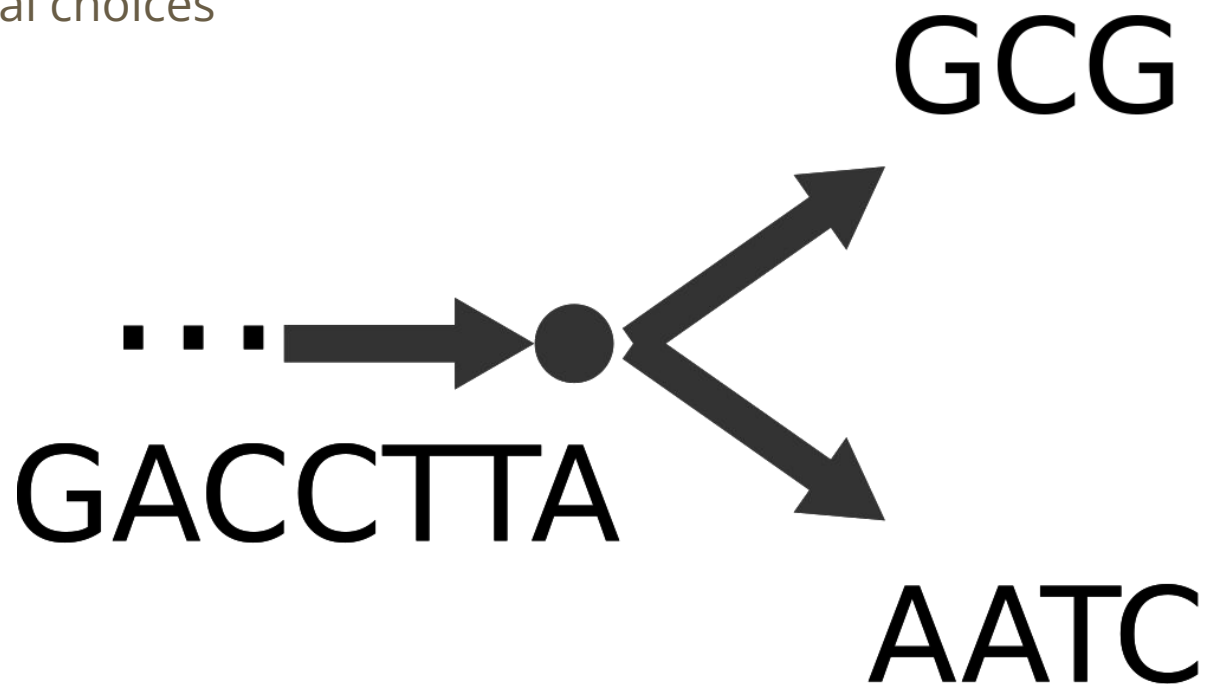
and in the initial molecules, an exon is shared:

**exon a** **exon b**

**exon a** **exon c**

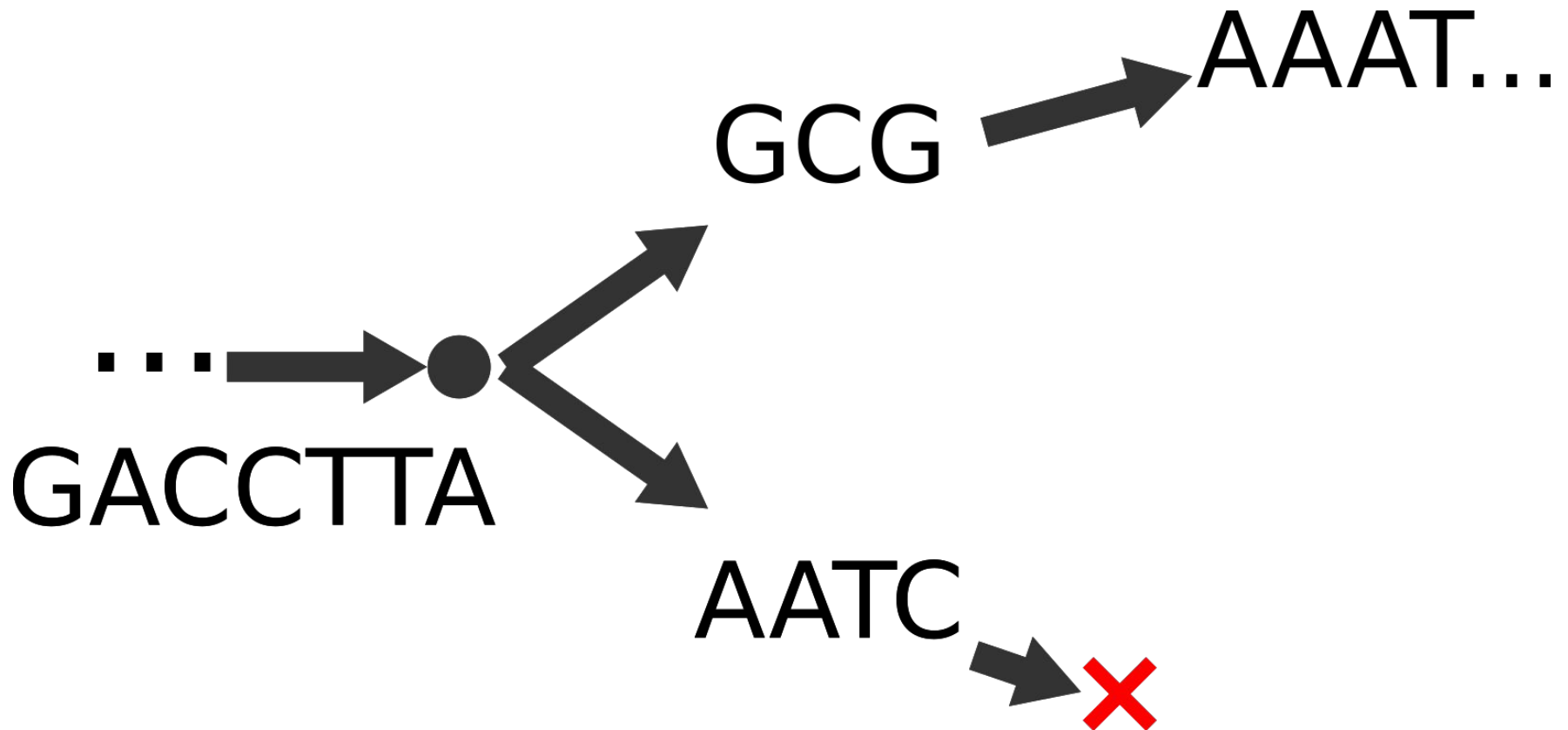
# Greedy algorithms

local choices



# Greedy algorithms

local choices can lead to bad decisions

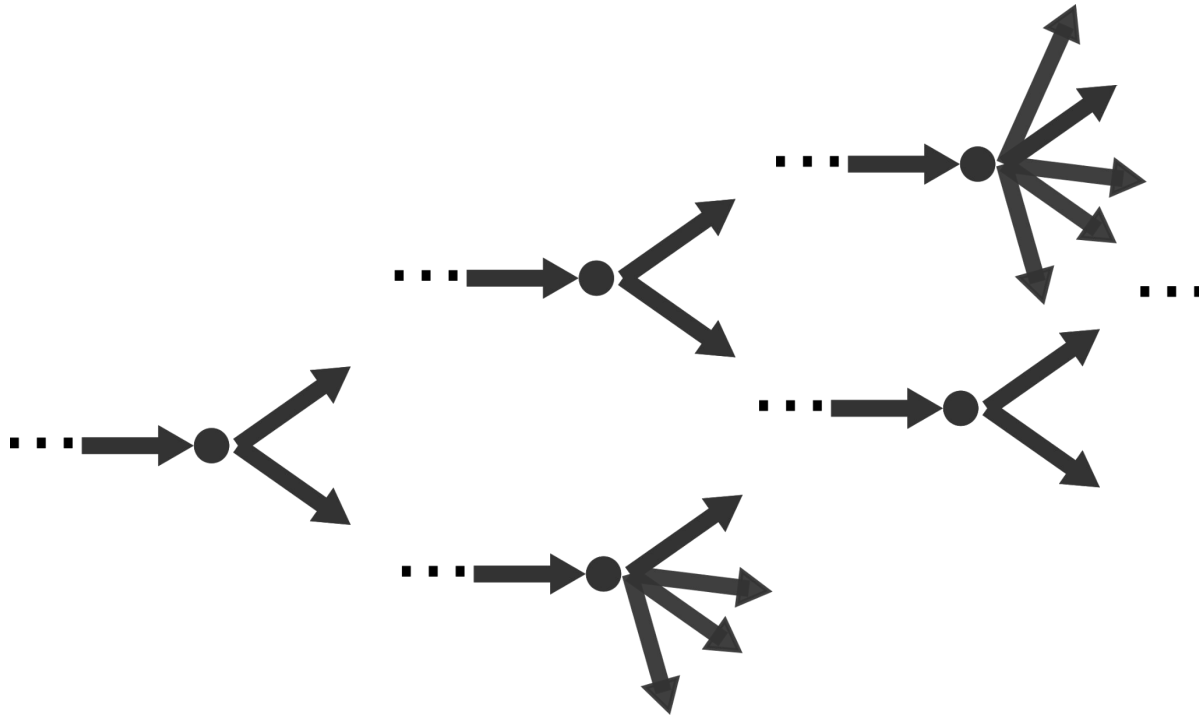




# All vs all overlaps algorithms

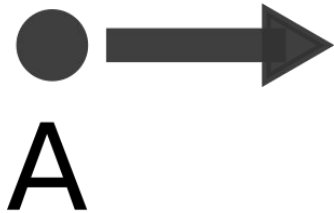
Have a global view of the possibilities in the “maze”

Ideal but... **quadratic**



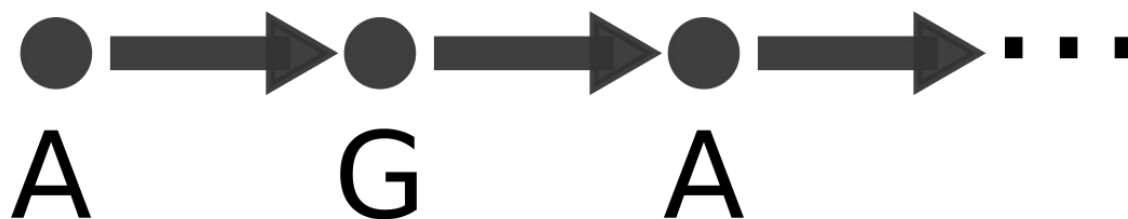
# de Bruijn graph assembly

With de Bruijn graphs we walk in the maze nucleotide by nucleotide:



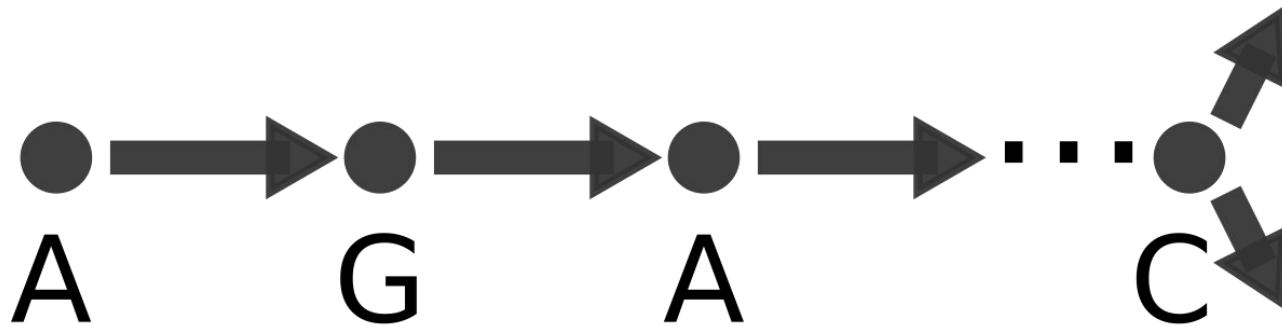
# de Bruijn graph assembly

Your next step must correspond to the nucleotide that comes after in the original transcript



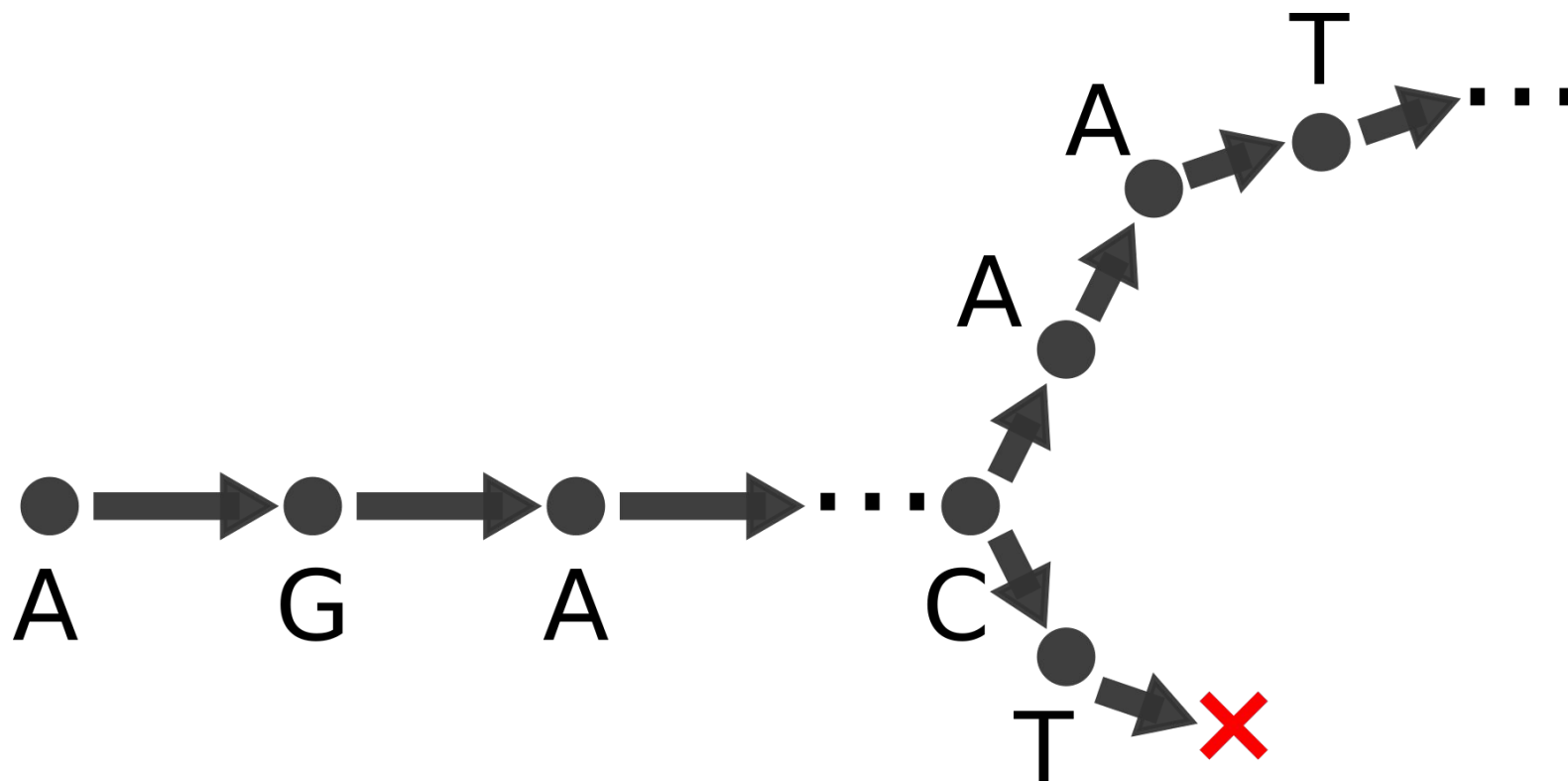
**Result:** concatenation of the nucleotides (AGA...)

# de Bruijn graph assembly



# de Bruijn graph assembly

Some dead ends and other bifurcations can be seen



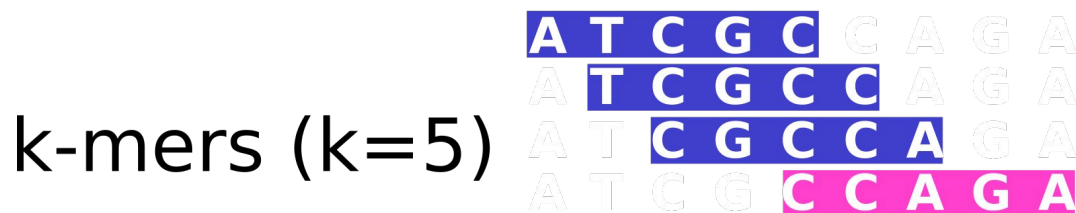
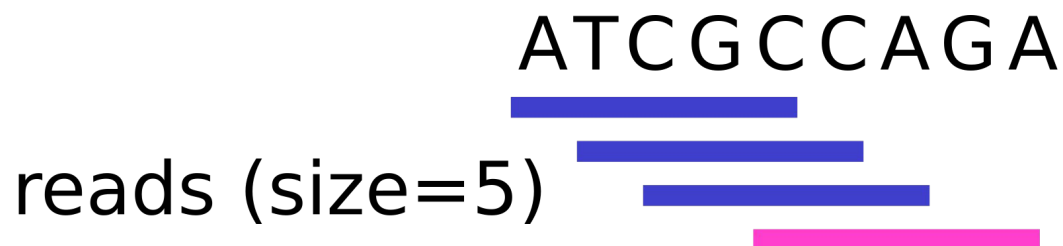
# de Bruijn graph assembly

Store the “maze” in a graph structure (de Bruijn graph)

- ❑ helps with local choices
- ❑ cost efficient (RAM & runtime)

# de Bruijn graph in practice: k-mers

k-mers: why don't we use reads



result: ATCGCCA, CCAGA

# de Bruijn graph in practice: k-mers

k-mers (k=4)

A T C G C C A G A  
A T C G C C A G A  
A T C G C C A G A  
A T C G C C A G A  
A T C G C C A G A  
A T C G C C A G A

result: ATCGCCAGA



# de Bruijn graph in practice: k-mers

k-mers help bridging the assembly

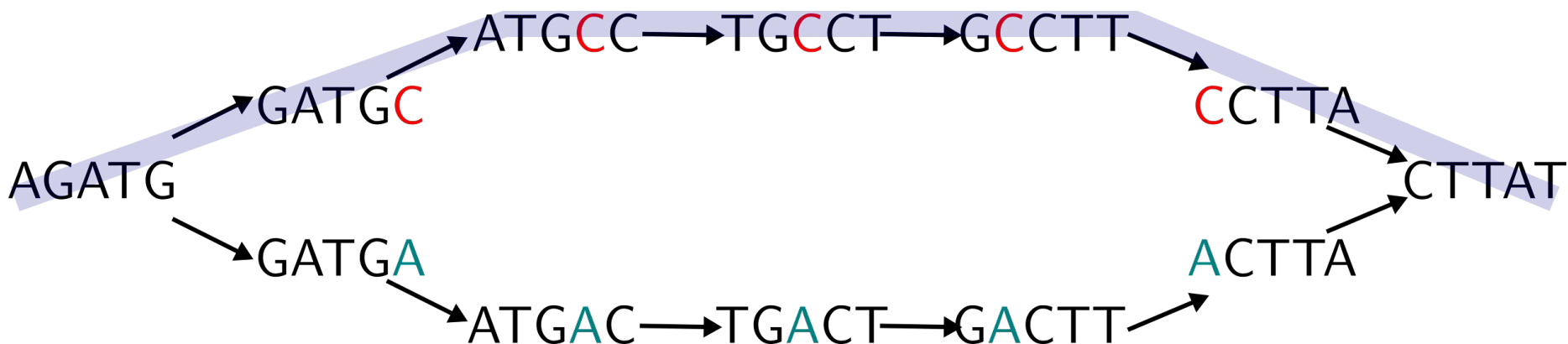
they are key elements to work with the dBG

in practice implementations allow using several k sizes

tradeoff larger k: more conservative /smaller k: more gaps filled in the graph

# Path in the De Bruijn graph

De Bruijn graph



assembly : a set of gap-less sequences extracted from paths covering the graph (after some modifications to the graph...)

# Vocabulary: bubbles/bulges

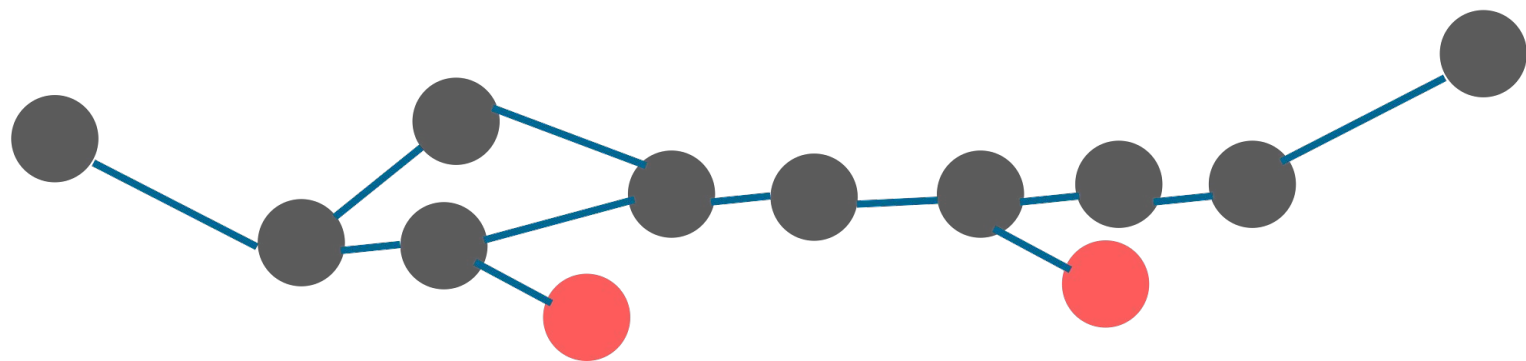
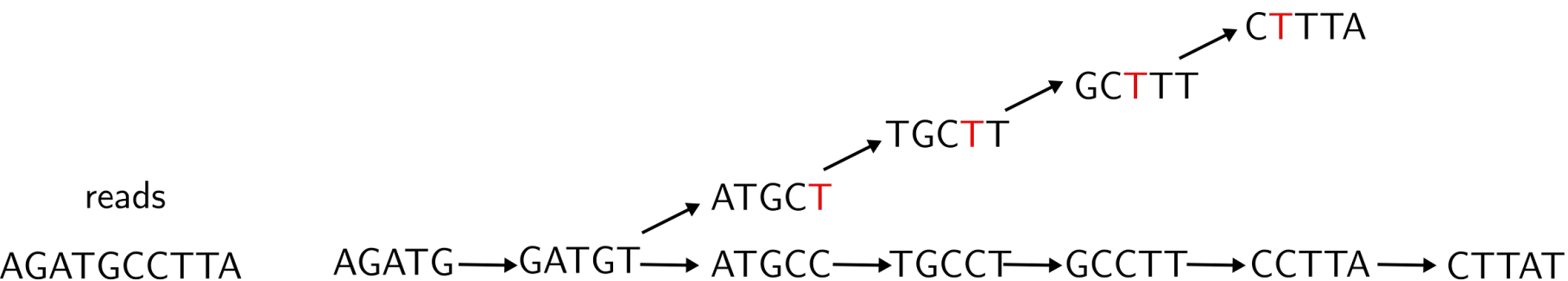
AGATGCCTTAT

AGATG → GATGC → ATGCC → TGCCT → GCCTT → CCTTA → CTTAT

AGATGCCTTAT  
AGATGACTTAT

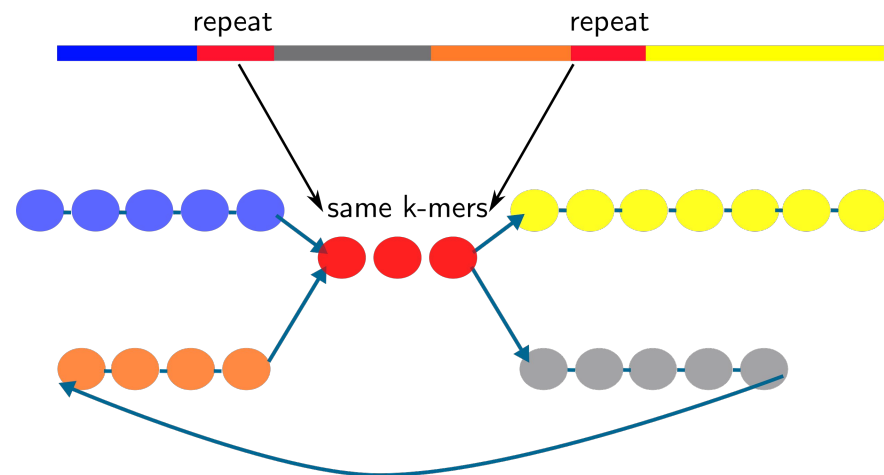
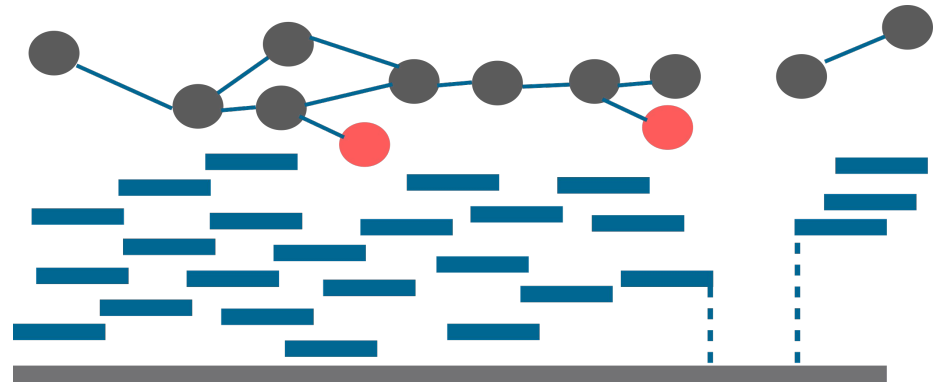


# Vocabulary: tips/dead ends



# An assembly generally is

- smaller than the reference,
- fragmented
- missing reads create gaps
- repeats fragment assemblies and reduce total size



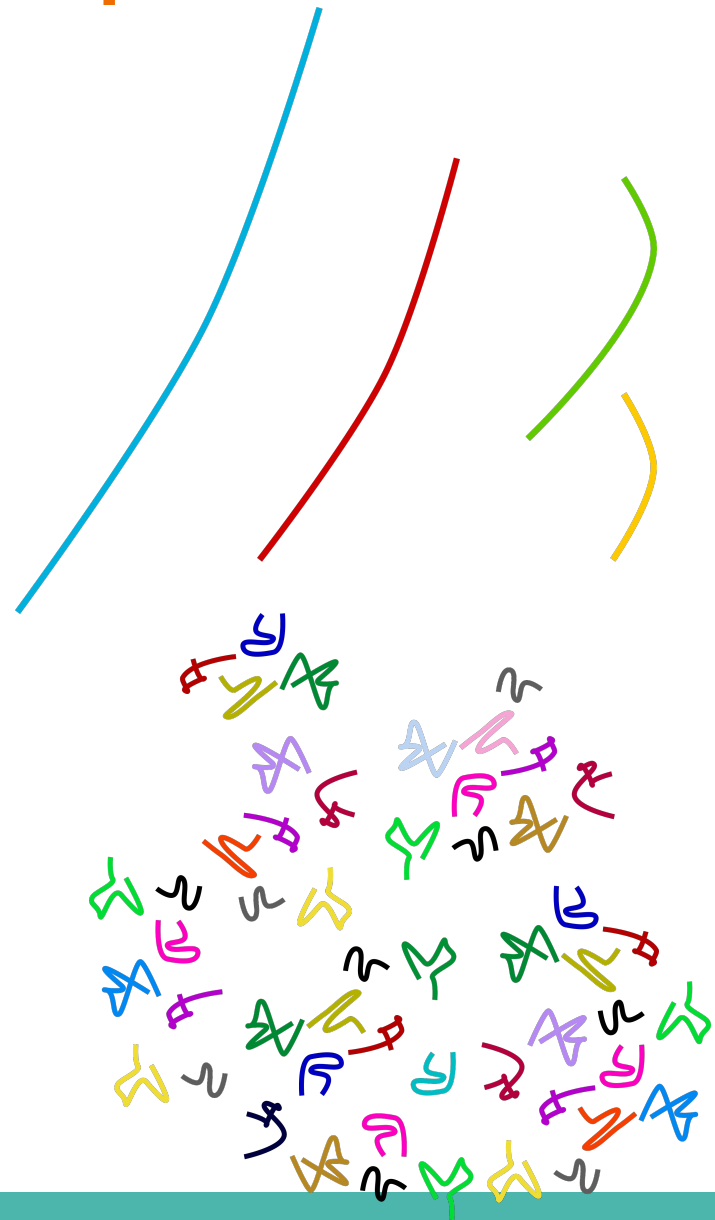
# Contrasting genome and transcriptome assemblies

## genome

- uniform coverage
- single contig per locus
- double stranded
- theory: one massive graph per chromosome
- practice: repeats aggregate, contigs smaller than chromosomes

## transcriptome

- exponentially distributed coverage
- multiple contigs per locus
- strand specific
- theory: thousands of small disjoint graphs, one per gene
- practice: gene families, ALU & TE, low covered

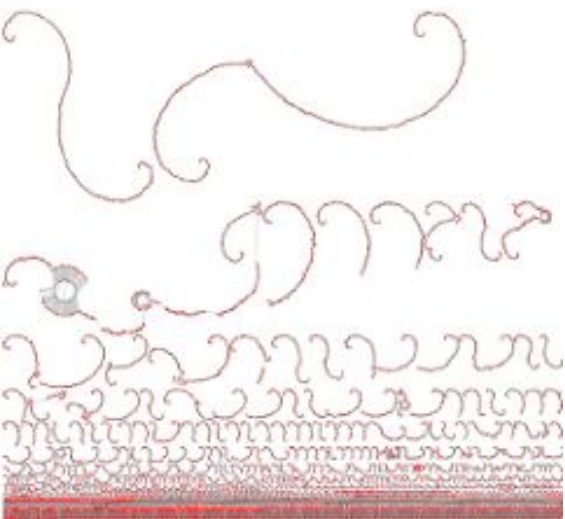


# Contrasting genome and transcriptome assemblies

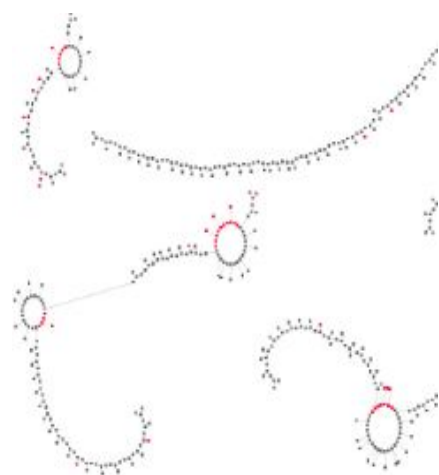
Despite these differences, DNA-seq assembly methods apply:

- Construct a de Bruijn graph (same as DNA)
- Output contigs (same as DNA)
- Allow to re-use the same contig in many different transcripts (new part)

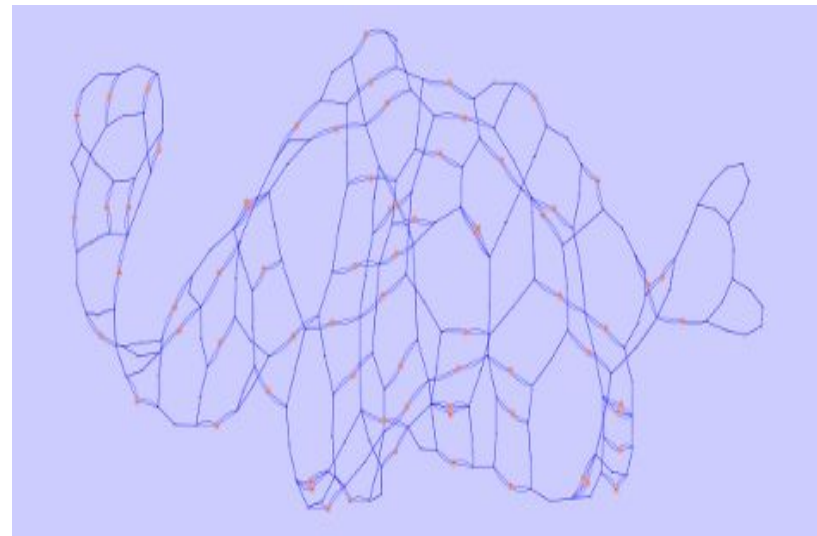
# Real instance graphs



graph from shallow covered Drosophila dataset



zoomed-in bubbles (+ tips)



gene family

Credit: ERABLE team (Lyon)



# There is no single solution for assembly...

Conclusions of the GAGE benchmark : in terms of assembly quality, there is no single best assembler. Applies to RNA-seq.

Main tools:

- TransAbyss**, Robertson et al. *Nat. Met* 2010 <https://github.com/bcgsc/transabyss>
- Bridger**, Chang et al. *Genome Biol.* 2015 [https://github.com/fmaguire/Bridger\\_Assembler](https://github.com/fmaguire/Bridger_Assembler)
- SOAPdenovo-Trans**, Xie et al. *Bioinformatics* 2014  
<https://github.com/aquaskyline/SOAPdenovo2>
- Trinity**, Grabherr et al. *Nat. Biotechnol.* 2011  
<https://github.com/trinityrnaseq/trinityrnaseq/wiki>
- **rnaSPAdes**, Bushmanov et al. *GigaScience* 2019 <http://cab.spbu.ru/software/spades/>

# The main building blocks in theory

1. (optional) correct the reads (for instance BayesHammer in rnaSPAdes)
2. build a graph from the reads (remove k-mers seen once)
3. remove likely sequencing errors (tips)
4. remove known patterns (bubbles)
5. return simple paths (i.e. contigs), **allow nodes to be used several times**

# Warning: what's in the paper is different than what's in the implementation...

## 2. Assembly in SPAdes: An Outline

Go to:

Below we outline the four stages of SPAdes, which deal with issues that are particularly troublesome in SCS: sequencing errors; non-uniform coverage; insert size variation; and chimeric reads and bireads:

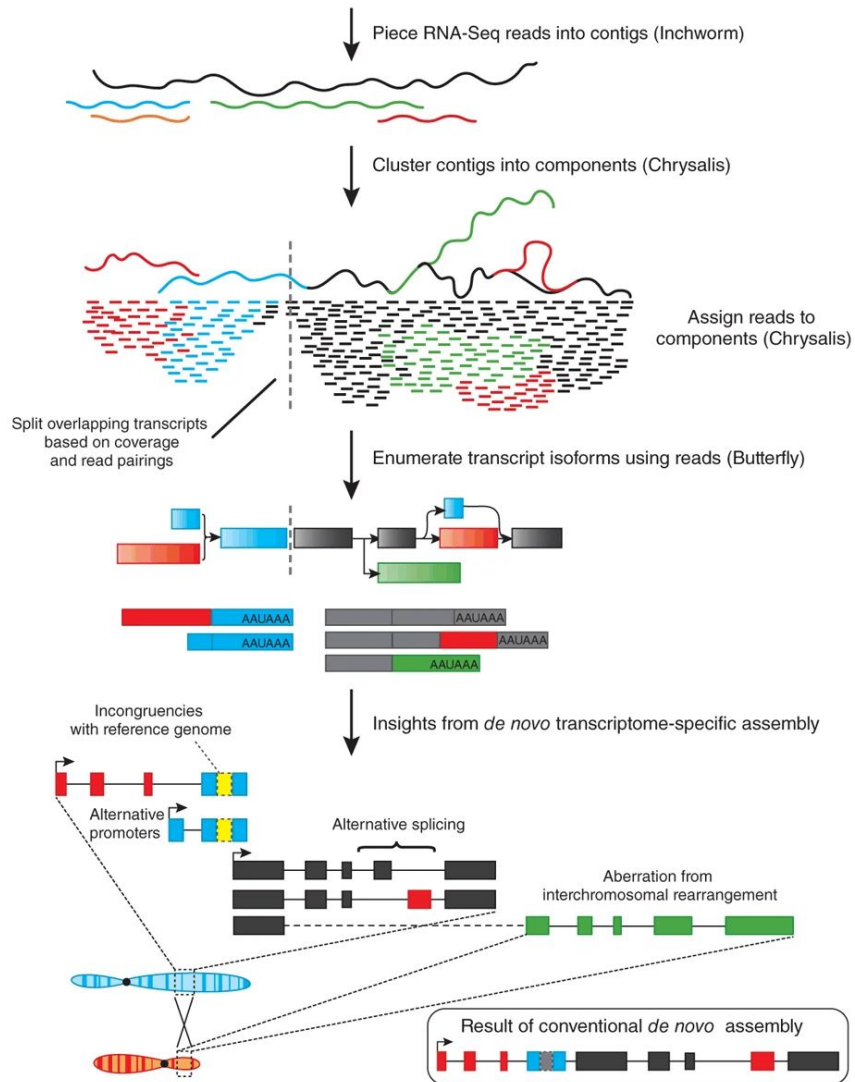
- (1) Stage 1 (assembly graph construction) is addressed by every NGS assembler and is often referred to as de Bruijn graph *simplification* (e.g., *bulge/bubble* removal in EULER/Velvet). We propose a new approach to assembly graph construction that uses the *multisized de Bruijn graph*, implements new bulge/tip removal algorithms, detects and removes chimeric reads, aggregates biread information into *distance histograms*, and allows one to backtrack the performed graph operations.
- (2) Stage 2 (**k-bimer adjustment**) derives accurate distance estimates between *k*-mers in the genome (edges in the assembly graph) using joint analysis of distance histograms and paths in the assembly graph.

# Trinity assembler



- Inchworm de Bruijn graph construction, part 1
- Chrysalis de Bruijn graph construction, part 2
- Butterfly Graph traversal using reads, isoforms enumeration

# Trinity overall



Iyer et al, Nature Biotech., 2011

# Trinity: detail

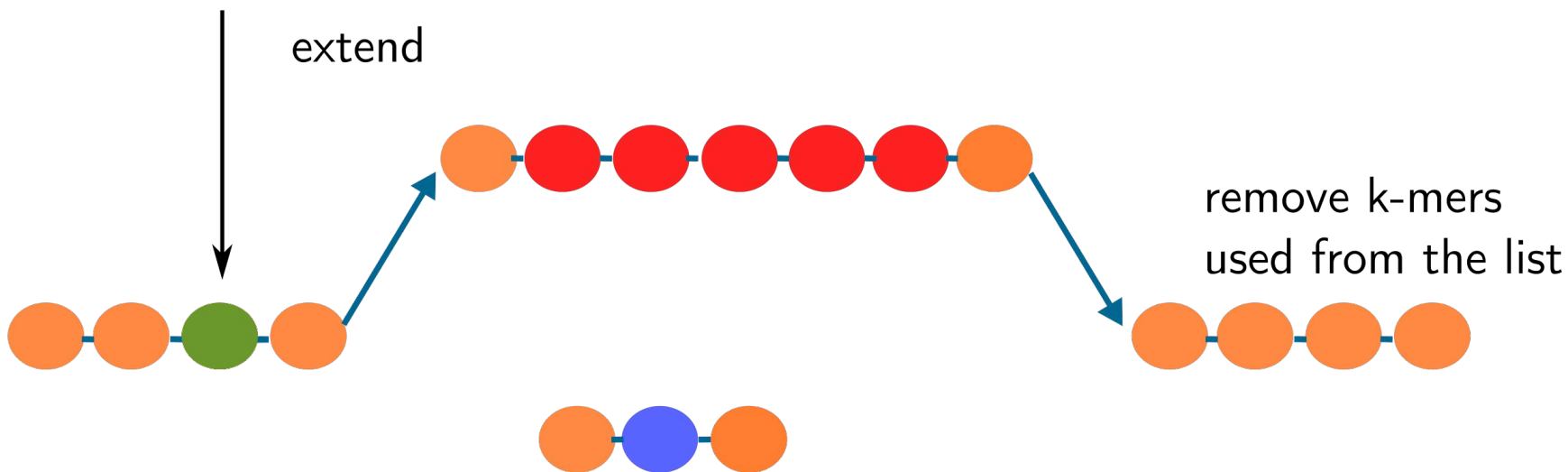
## 1-Inchworm

list all k-mers



● ● ● seed k-mers (high occurrence)

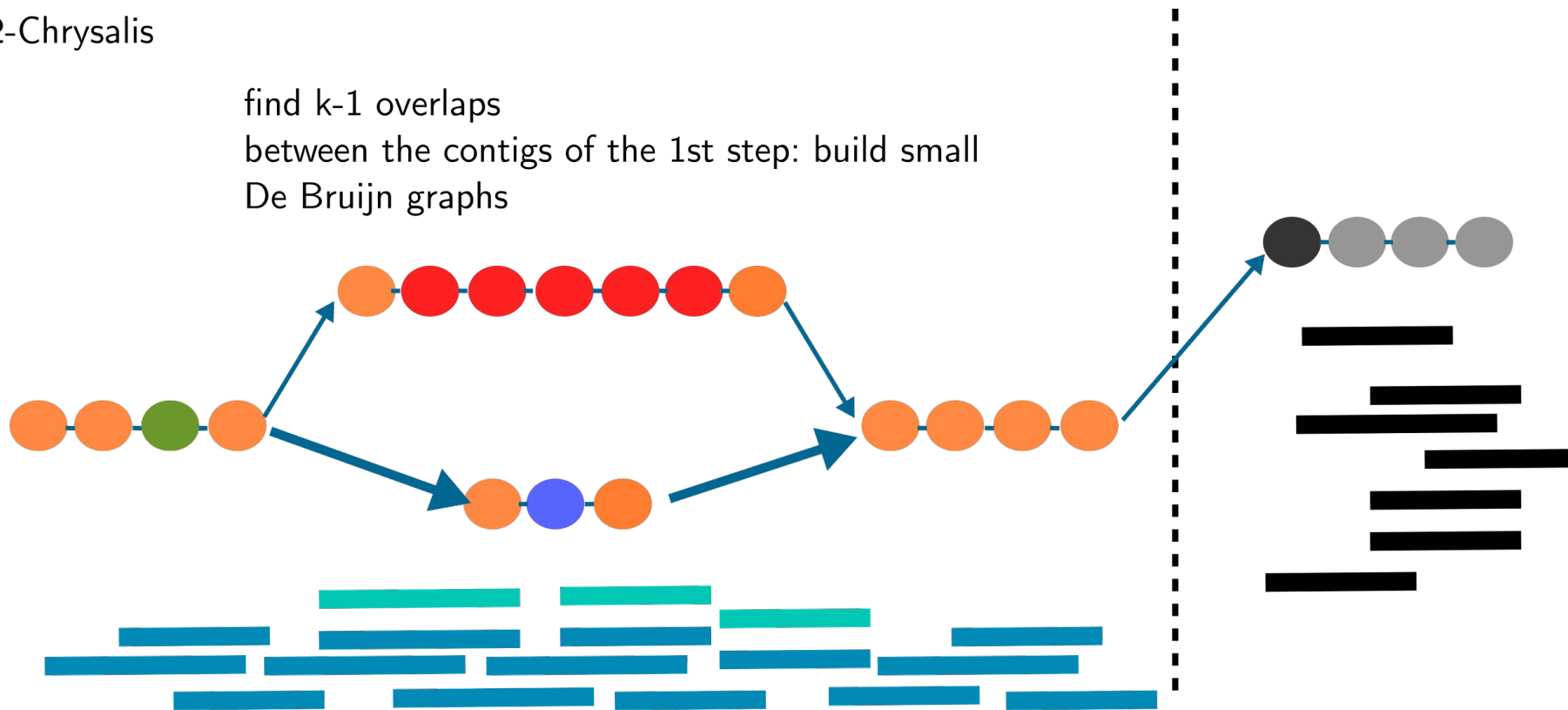
extend



# Trinity: detail

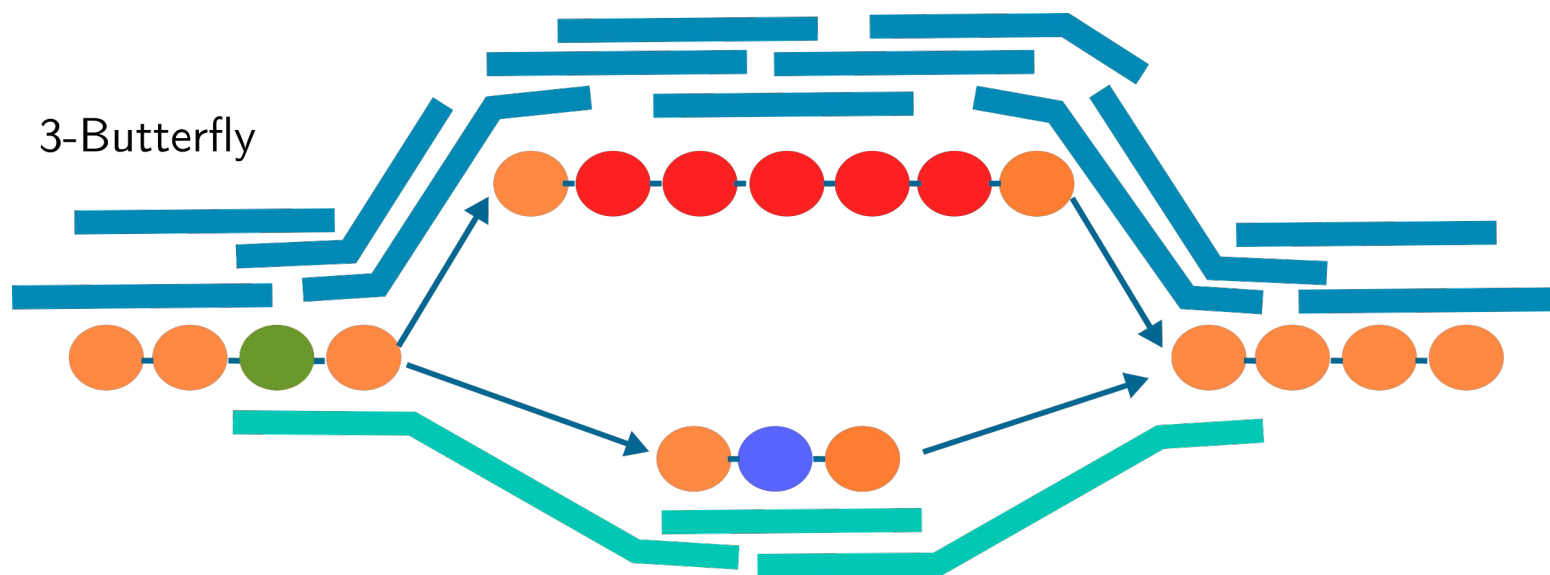
## 2-Chrysalis

find  $k-1$  overlaps  
between the contigs of the 1st step: build small  
De Bruijn graphs

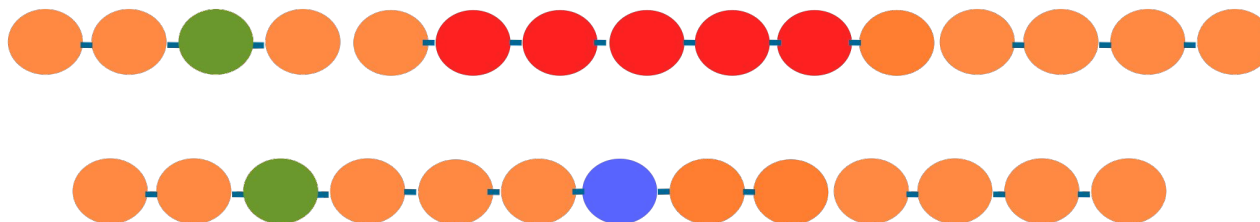


use read mapping information to separate clusters

# Trinity: detail



output read-coherent isoforms





# Trinity output

```
>TRINITY_DN1000_c115_g5_i1 len=247 path=[31015:0-148 23018:149-246]
```

```
AATCTTTTTTGGTATTGGCAGTACTGTGCTCTGGGTAGTGATTAGGGCAAAGAAGACAC
```

```
ACAATAAAGAACCAGGTGTTAGACGTCAGCAAGTCAAGGCCTTGGTTCTCAGCAGACAGA
```

```
AGACAGCCCTTCTCAATCCTCATCCCTTCCCTGAACAGACATGTCTTCTGCAAGCTTCTC
```

```
CAAGTCAGTTGTTACAGGAACATCATCAGAATAAATTTGAAATTATGATTAGTATCTGA
```

```
TAAAGCA
```

-Trinity read cluster 'TRINITY\_DN1000\_c115'

- gene 'g5'

- isoform 'i1'

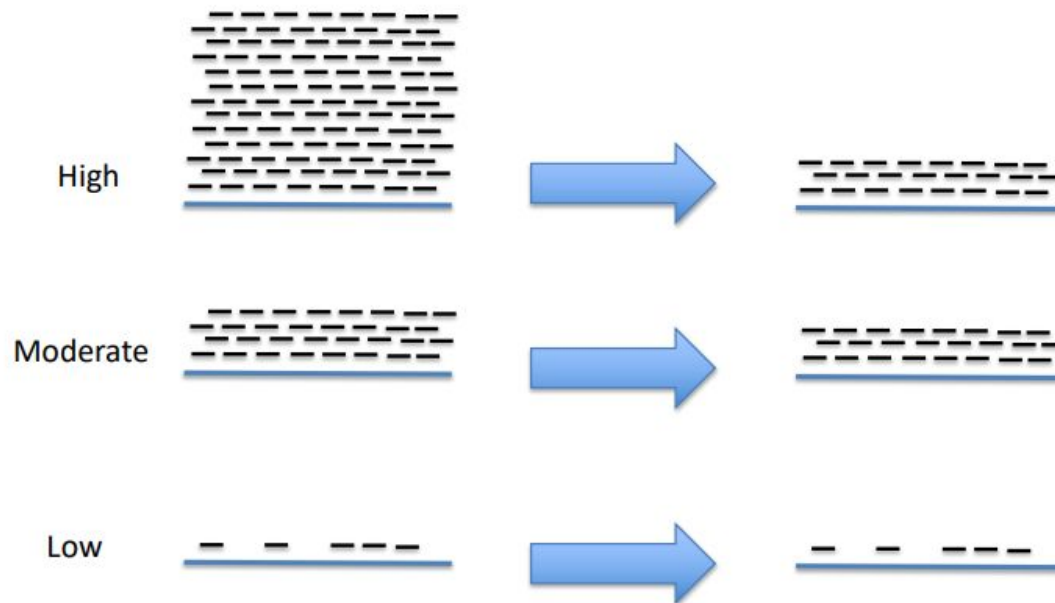
-path=[31015:0-148 23018:149-246]") indicates the path traversed in the Trinity de Bruijn graph to construct that transcript

# Normalization effects on assembly (example of Trinity)

From Brian

Haas

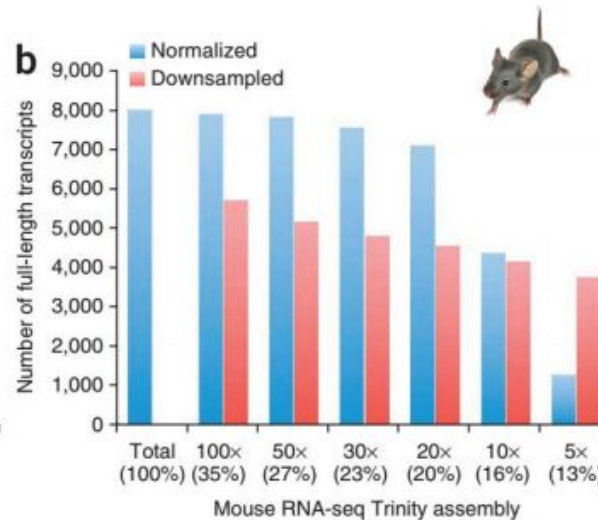
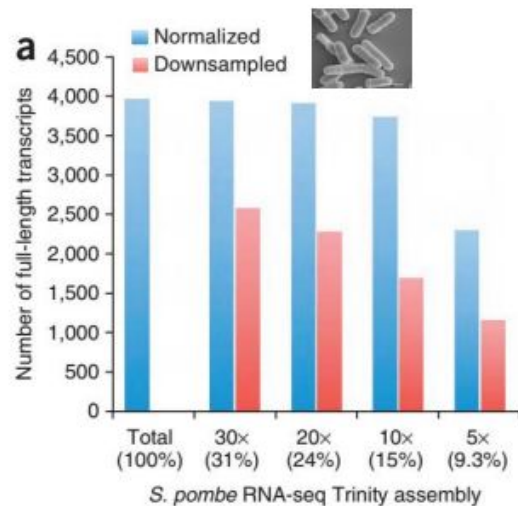
*In silico* normalization of reads



# Normalization effects on assembly (example of Trinity)

## Impact of Normalization on *De novo* Full-length Transcript Reconstruction

From Brian Haas

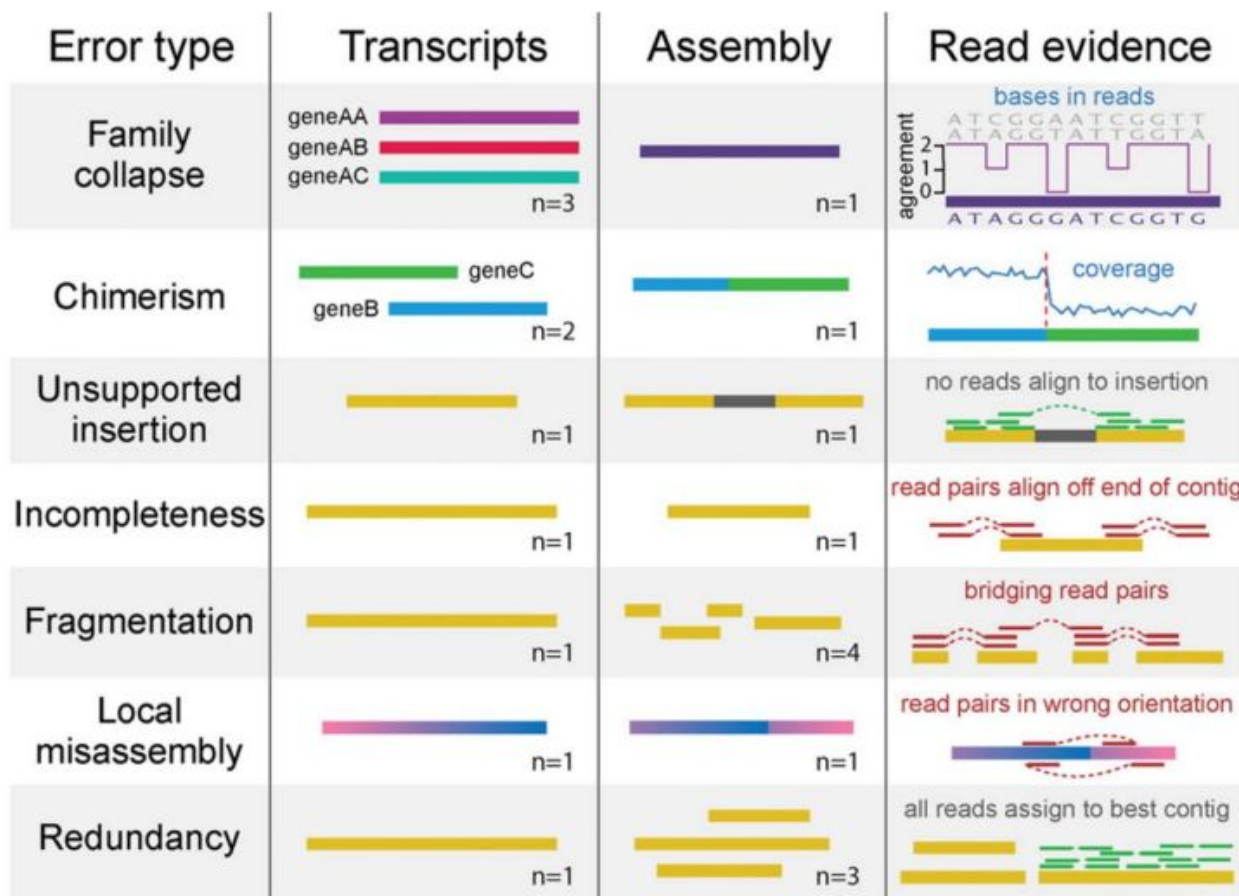


Largely retain full-length reconstruction, but use less RAM and assemble much faster.

Total (100%) 30x (31%) 20x (24%) 10x (15%) 5x (9.3%)  
*S. pombe* RNA-seq Trinity assembly

Total (100%) 100x (35%) 50x (27%) 30x (23%) 20x (20%) 10x (16%) 5x (13%)  
 Mouse RNA-seq Trinity assembly

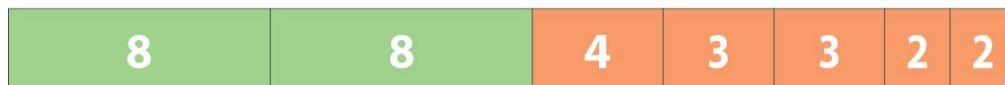
# Errors made by assemblers



Smith-Unna et al. Genome Research, 2016

# RNA-seq Assembly quality assessment

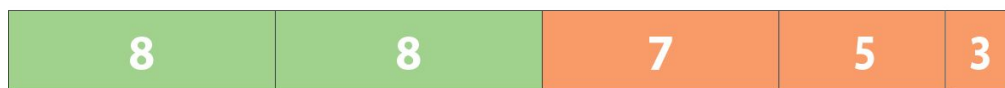
- N50 is not very useful.
  - unreasonable isoform annotation for long transcripts drives higher N50
  - very sensitive reconstruction for short lowly expressed transcripts leads to lower N50



N50 = 8

Average = 4.3

Mediane = 3



N50 = 8

Average = 6.2

Mediane = 7

Reference-free evaluation must be preferred, based on **read remapping**

Main tools:

- rnaQuast <http://cab.spbu.ru/software/rnaquast/>
- Transrate <http://hibberdlab.com/transrate/>



# TransRate

## 1 input data

assembled contigs    paired-end reads



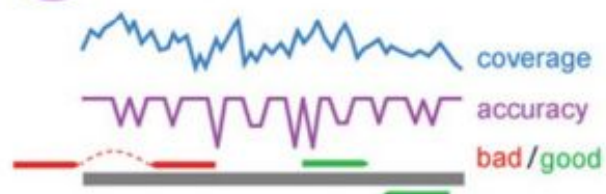
## 2 align reads to contigs



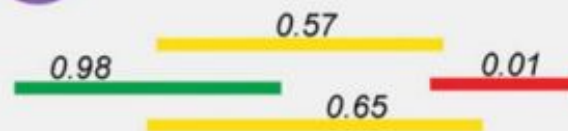
## 3 assign multimapping reads



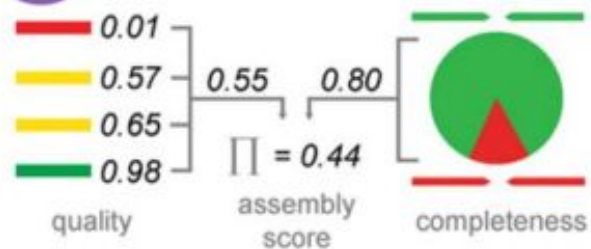
## 4 collect contig score components



## 5 calculate contig scores



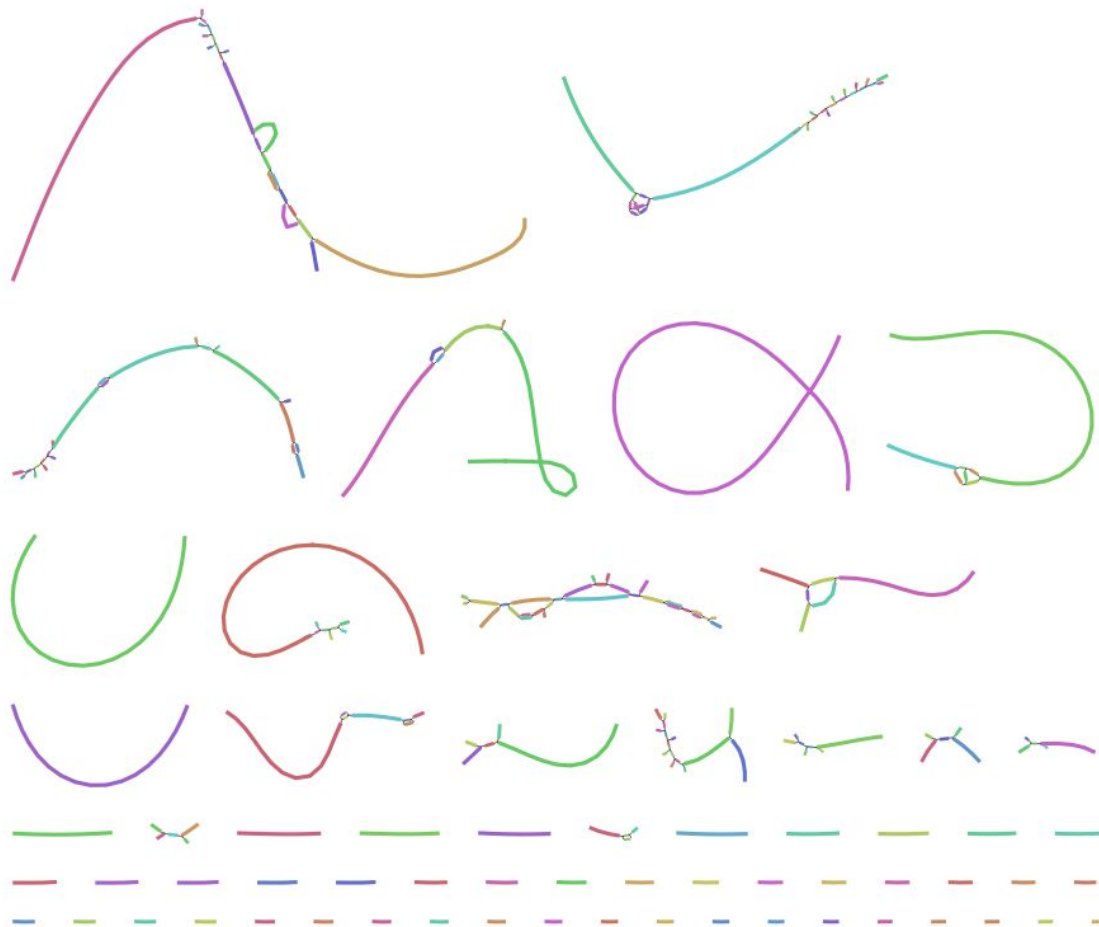
## 6 calculate assembly score



Smith-Unna et al. Genome Research, 2016

# Visualization: Bandage

<https://rrwick.github.io/Bandage/>



# Meta-practices

- 1- Read surveys, Twitter, blogs
2. Pick two assemblers
3. Run each assembler at least two times (different parameters)
4. Compare assemblies
5. If possible, visualize them

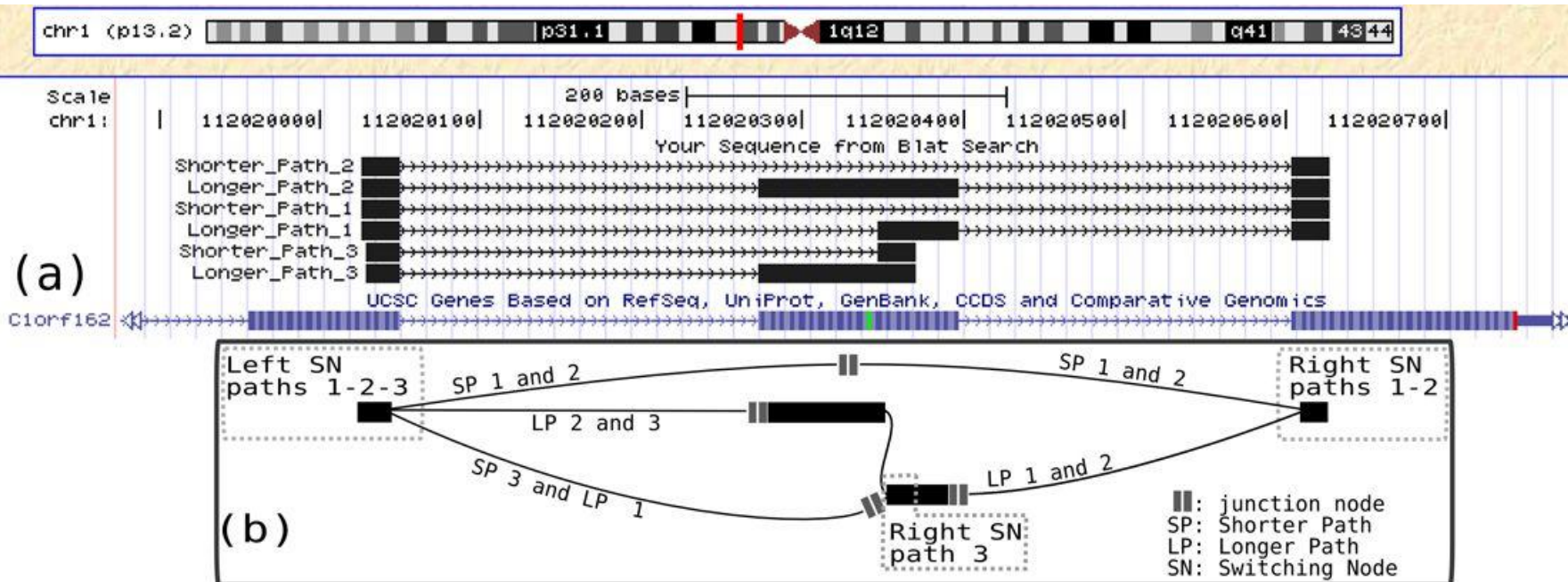
An assembly is not the absolute truth, it is a mostly complete, generally fragmented and mostly accurate hypothesis

Currently, Trinity, RNASpades and TransAbyss could be pointed as the most trustworthy/qualitative (for known species. Not one tool for all issues).



# Practical: Trinity assembly

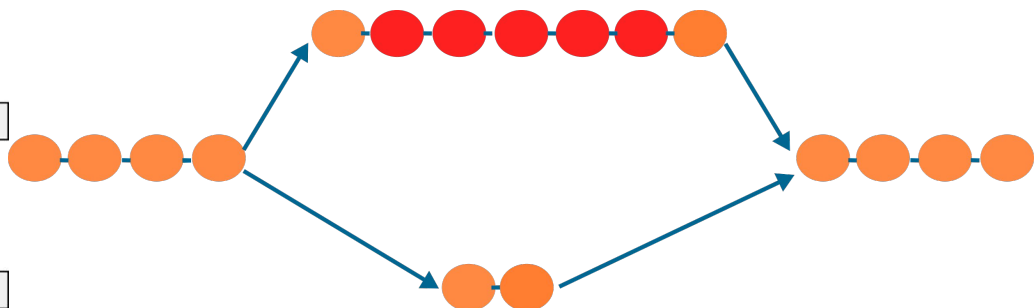
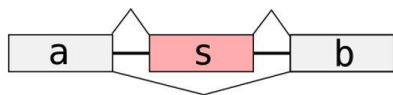
# Assembly does not output all variants



# KISSPLICE

Goal: instead of assembling full-length transcripts, KISSPLICE (Sacomoto et al. 2012) focuses on assembling ONLY the **bubbles** that contain events and **enumerate** the maximum of them

Exon Skipping



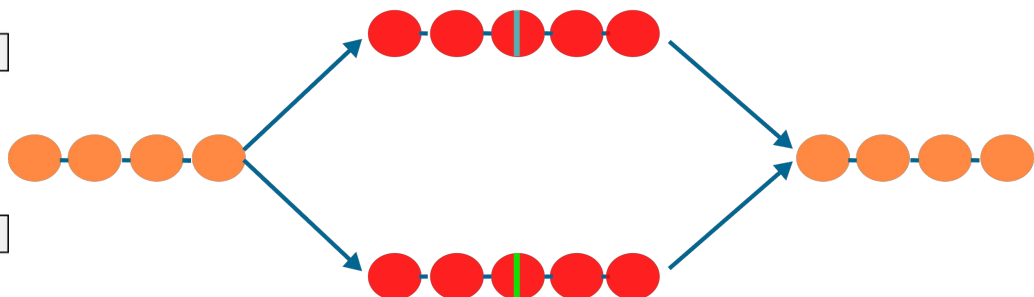
Intron retention



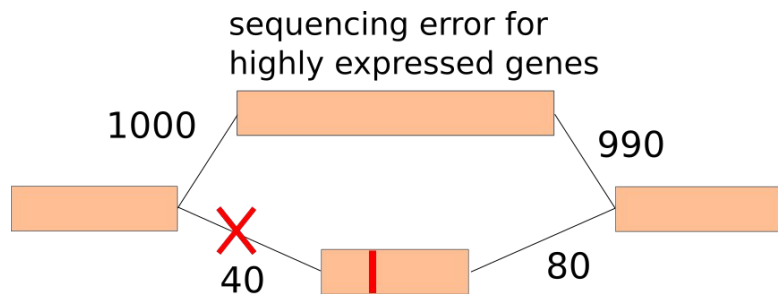
Alternative donor site



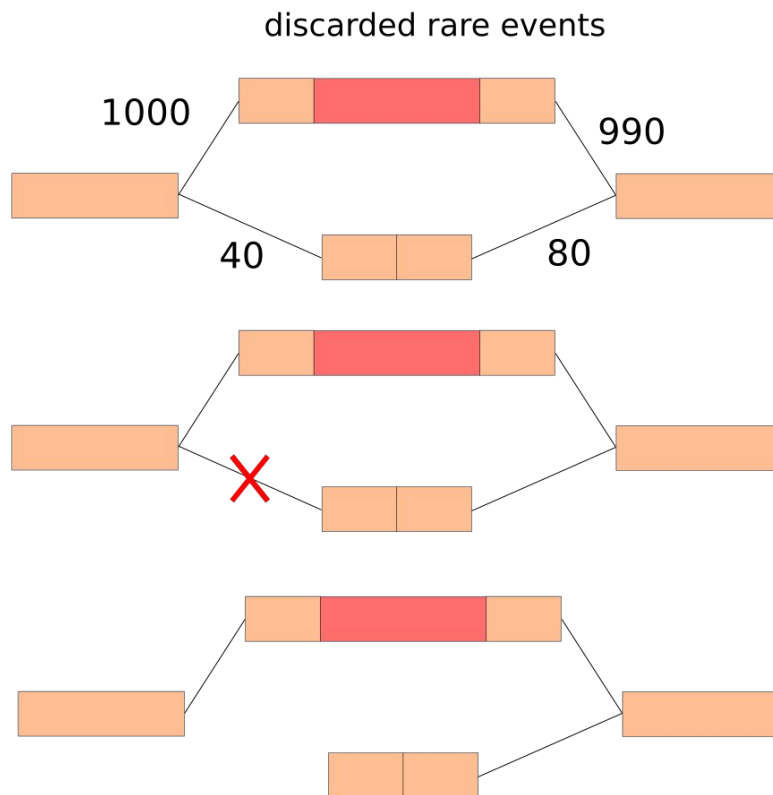
Alternative acceptor site



# KISSPLICE: graph cleaning + local assembly

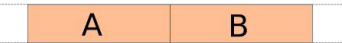


example: discard if ratio is  $< 0.05$



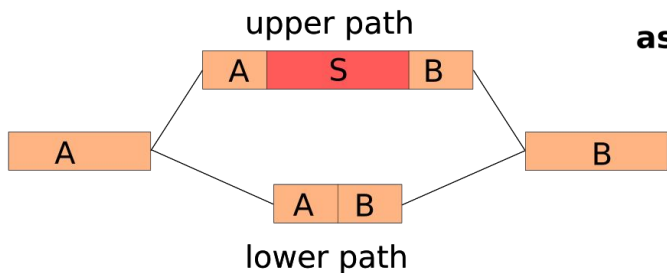
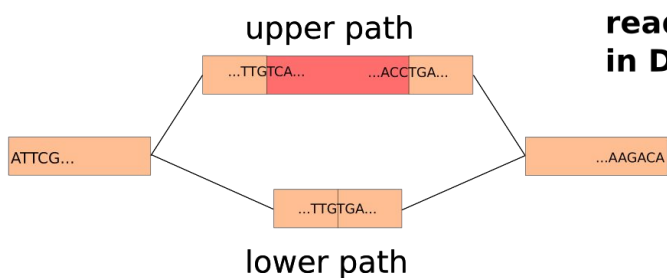
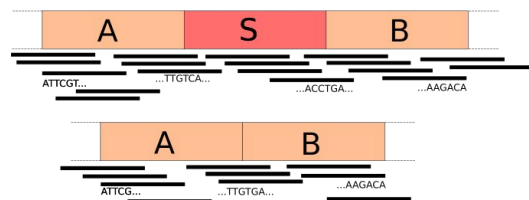
# Variants in local assembly

transcript 1 

transcript 2 

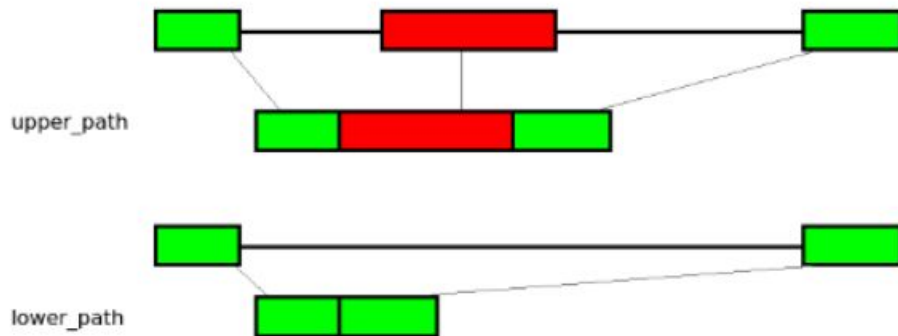
**local exon skipping**

**sequencing**



# KISSPLICE's output

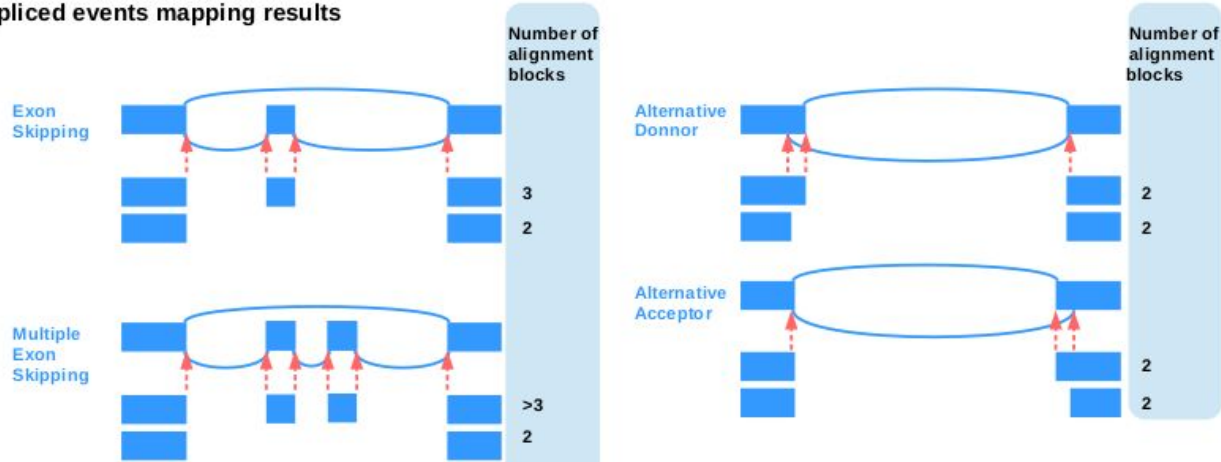
```
>bcc_89|Cycle_0|Type_1|upper_path_length_122|C1_0|C2_1|C3_2|C4_1|rank_0.55097  
CCCTGATGGCCTCAGAGGAGGAGTA AATGTGGGGACCTAGAGGAGGAGCTGAAAATTGTTACCAACAACCTTGAAATCCCTGGAGGCCAGGCGGACAAGTA TTCCACCAAAGAAGATAAATA  
>bcc_89|Cycle_0|Type_1|lower_path_length_46|C1_0|C2_0|C3_2|C4_6|rank_0.55097  
CCCTGATGGCCTCAGAGGAGGAGTA TTCCACCAAAGAAGATAAATA
```



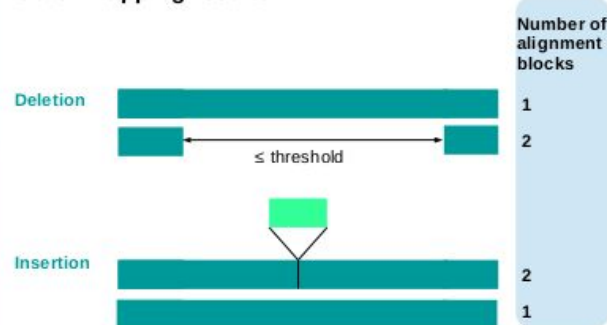
# Post-processings

What do I have?	What I can use	
I have a reference genome	<a href="#">KisSplice2refgenome</a>	differential analysis: <a href="#">kissDE</a>
I have no reference genome	<a href="#">KisSplice2refTranscriptome</a>	

## Spliced events mapping results

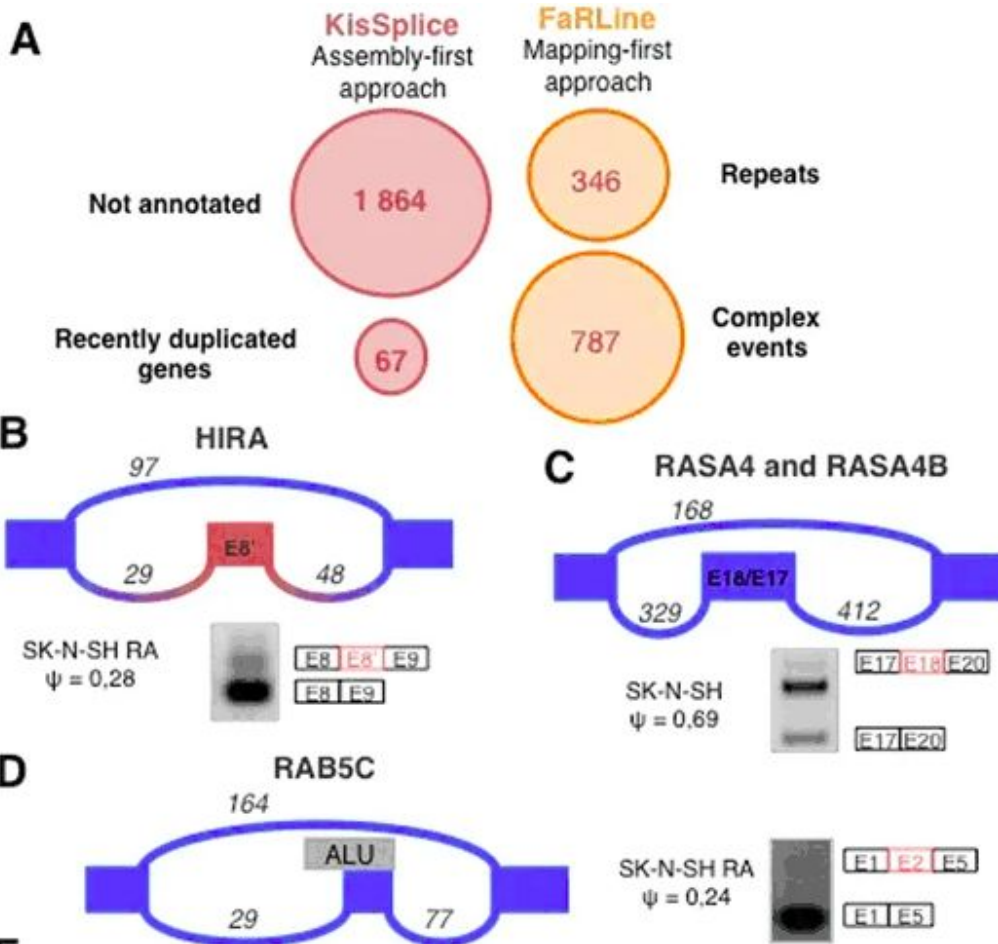


## Other mapping results



for quantification only  
see de-Kupl  
Audoux et al. 2017

# KISSPLICE case studies



**Discover splicing events:**  
Benoit Pilven et al. 2018

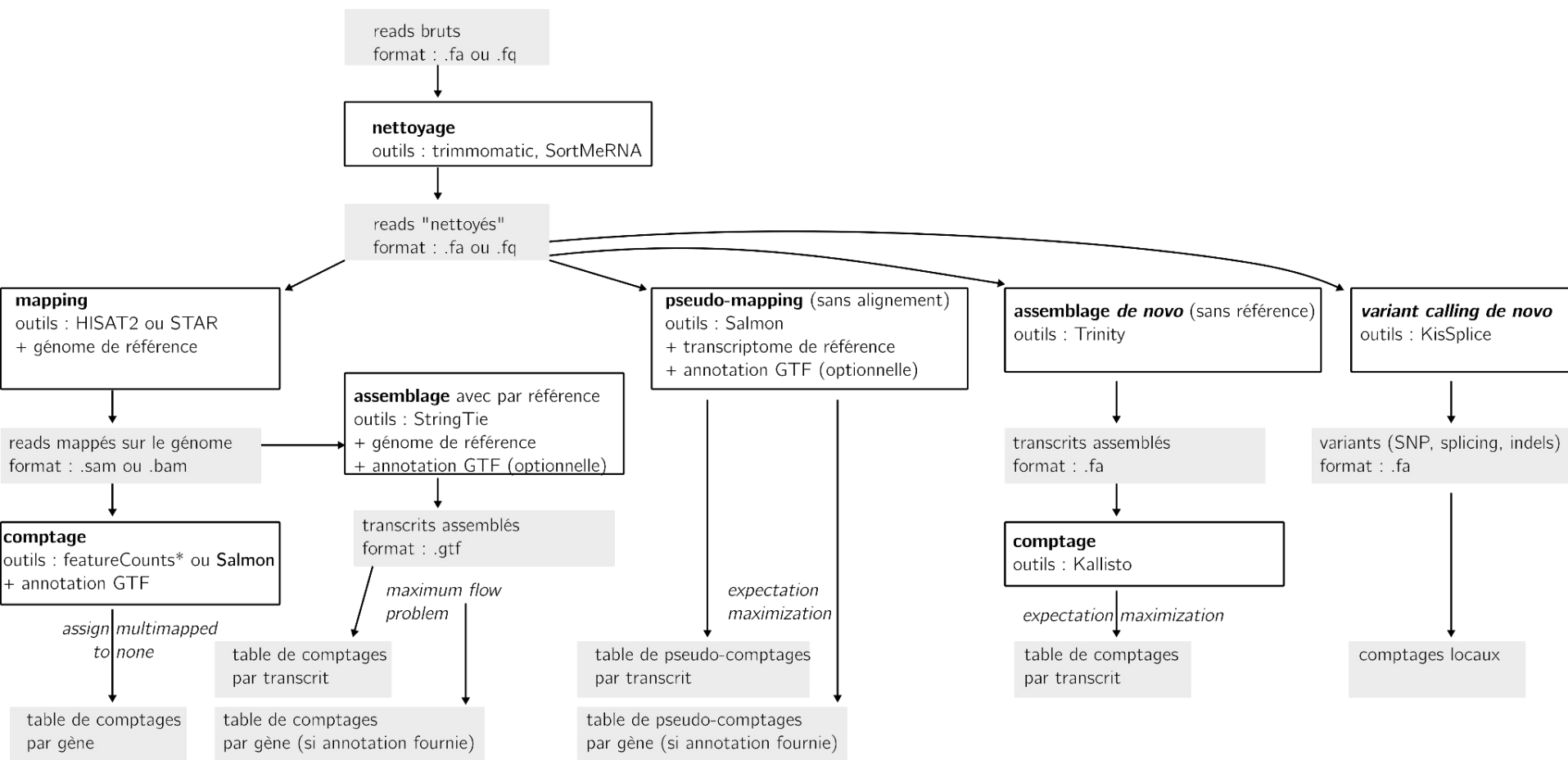
Farline: mapping  
**B** found only by Kisssplice (not annotated)  
**C** found only by Kisssplice (paralog)  
**D** found only by mapping (Alu repeat)

**Discover SNPs in pooled RNA-seq:** Lopez-Maestre et al. 2016



# Practical: Kissplice

# Our achievements (reference and *de novo*)



\* à vocation pédagogique pour le TP, déconseillé par ailleurs.

# Long reads : the ~~future~~ present of transcriptomics

Long reads overview

Possibilities & pipelines

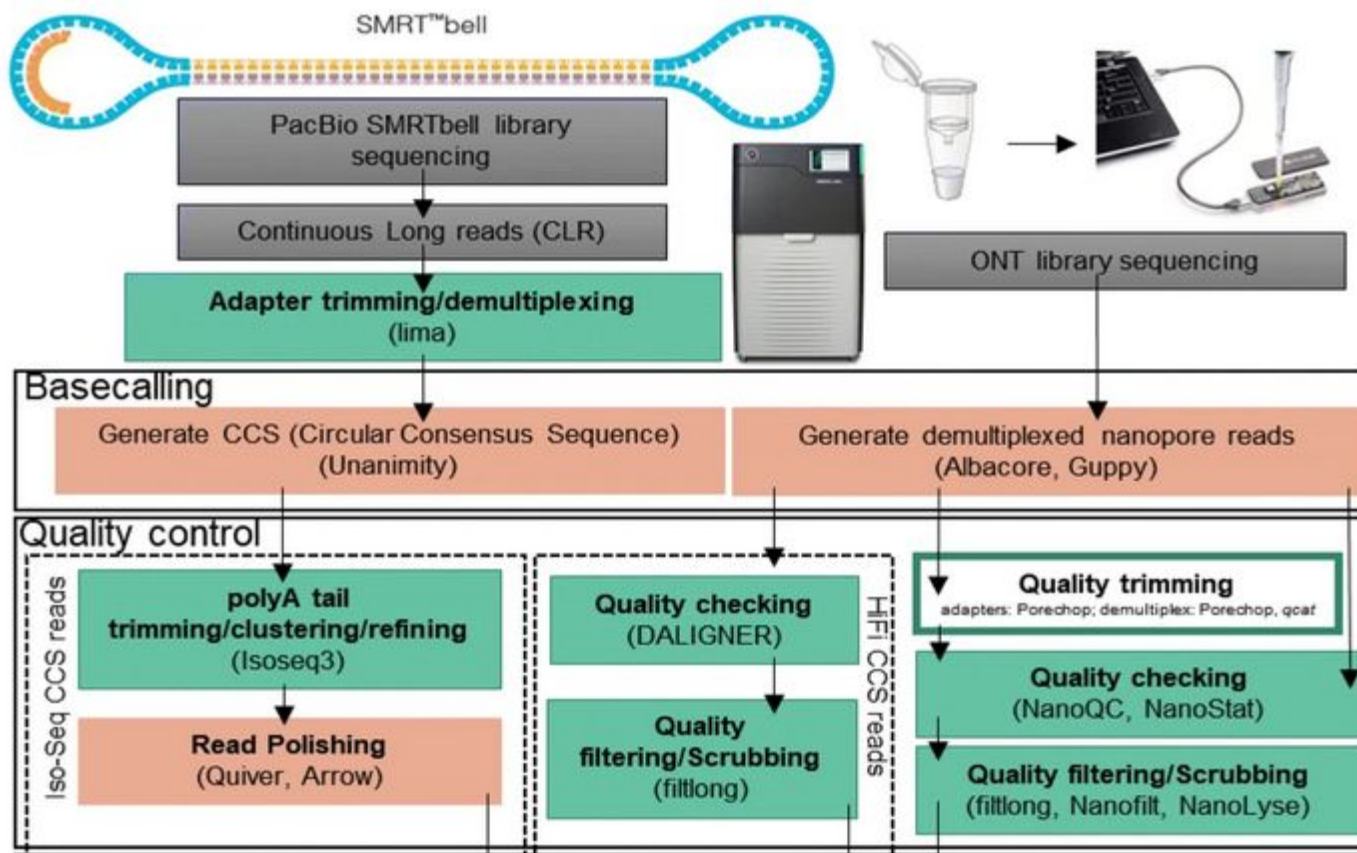
# Limitations of short reads

- ❑ recent studies suggest that our reference transcriptomes **miss isoforms**
- ❑ in particular in the context of **alternative splicing**
- ❑ *de novo* assembly of species with unknown/hardly known transcriptomes is still a challenge
- ❑ the mandatory cDNA step in short reads protocols implies **bias**

# Long reads technologies

- ❑ sequencing of long (>10kb) molecules is possible
  - ❑ **full RNAs!**
- ❑ with a higher (~1-5% to 14%) **error rate**
- ❑ **error profile** is different from SR: indels in **homopolymers**
- ❑ some allow to sequence directly RNA (reduced bias, epitranscriptomics)

# Long reads technologies



from Shanika L. Amarasinghe et al. Genome Biol. 2020

# Pacific Biosciences (Pacbio)

- ❑ in the case of RNA, a fragment is **read several times** and a consensus is computed
- ❑ read length limited by the longevity of the polymerase
- ❑ circular consensus sequence quality =  $f(\text{fragment length, pol longevity})$
- ❑ 4 passes : 1% error (0.1% reached after 9 passes)
- ❑ bias for indels in homopolymers

# Pacific Biosciences (Pacbio)

- ❑ the protocol is better suited for studying **isoform identification only** (not quantification)
  - ❑ initial overrepresentation of shorter molecules lead to size selection which introduces a bias
  - ❑ mitigation solutions still in progress



# Oxford Nanopore technologies (ONT)

- ❑ no limit to read length
- ❑ the fragment is read only once in the pore
- ❑ read quality depends on the speed of the fragment through the pore
  - ❑ **quality decreases in the late stages** of sequencing
- ❑ error rate >5%
- ❑ bias for **indels in homopolymers**

# Oxford Nanopore technologies (ONT)

- ❑ 1D sequencing protocol : **single pass** of strands
- ❑ (1D<sup>2</sup> protocol: sequence the **complementary strand immediately after** the forward strand and compute a consensus)
- ❑ accuracy over homopolymers is in progress (from R10 chemistry)

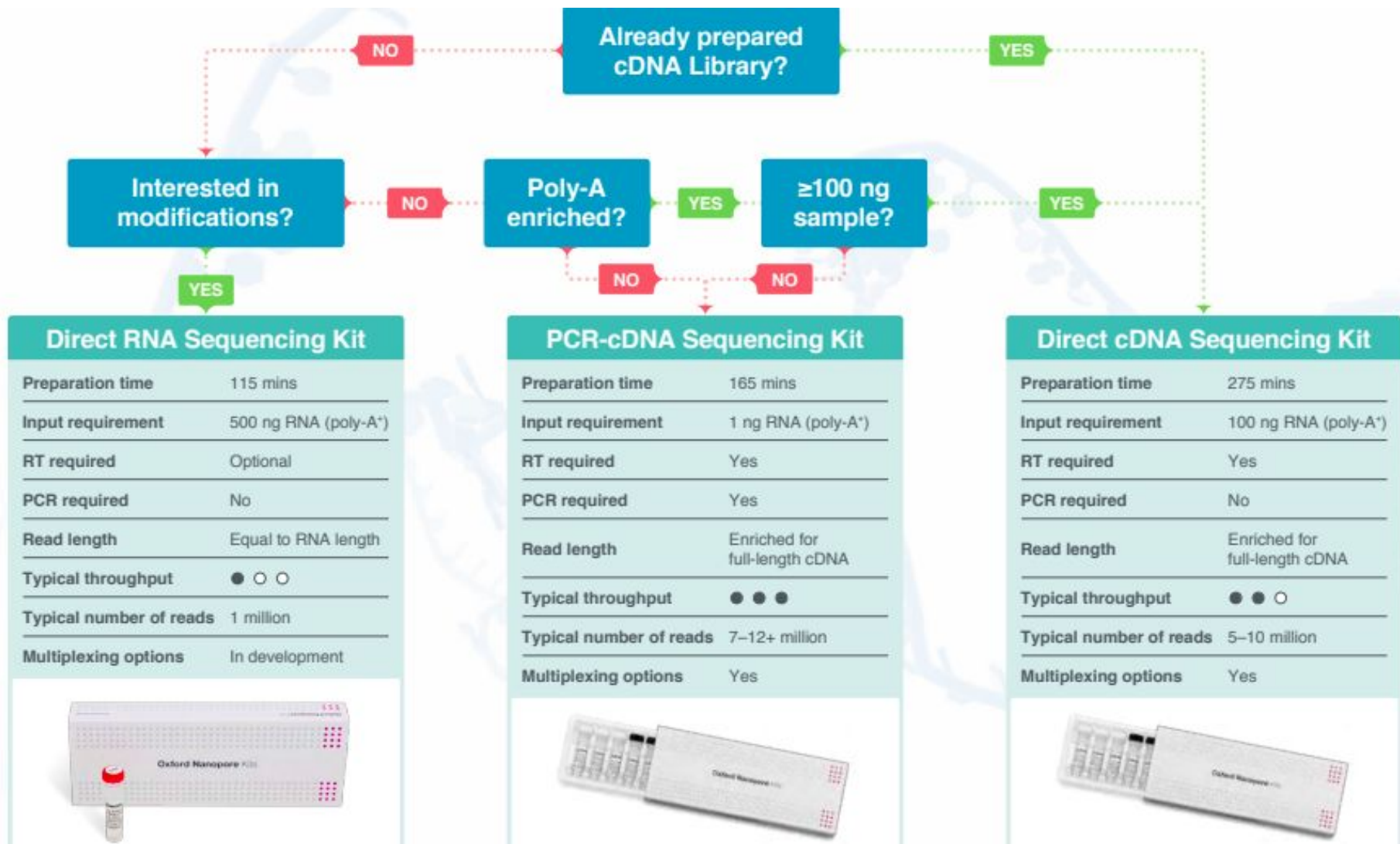
# Oxford Nanopore technologies (ONT)'s RNA direct

Methods based on reverse transcription:

- ❑ Template switching and artifactual splicing
- ❑ Loss of strandedness information
- ❑ Loss of base modifications
- ❑ Propagation of error due to PCR

Direct RNA

- ❑ no bias due to PCR
- ❑ possible to study some RNA modifications
- ❑ as of today not adequate for quantification (too much material is required)



material from Oxford Nanopore

# What has been studied with long reads so far

Near mature:

- ❑ **Quantification** of already **known genes** and isoforms
- ❑ **Quantification** of **novel isoforms** from known genes
- ❑ Transcript reconstruction (assembly) **based on a reference**

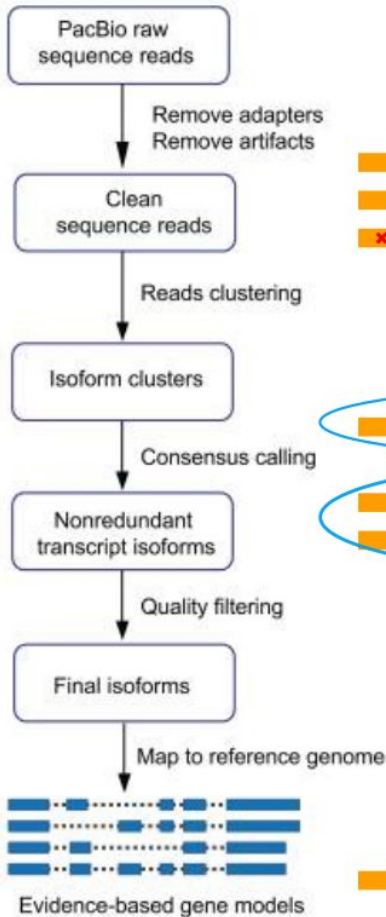
# What has been studied with long reads so far

## Exploratory:

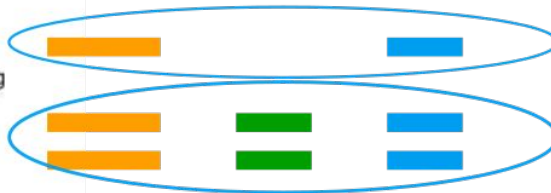
- ❑ RNA of paralogous genes (Dougherty et al., 2018, Chen et al., 2017)
- ❑ Fusion transcripts (Nattestad et al., 2018).
- ❑ Allele-specific expression (Tilgner et al., 2014), avelier et al., 2015).

# Spirit of most analysis pipelines

## Informatics pipeline

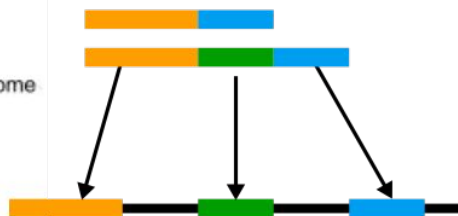


reads  
comparison all vs all



clusters:  
isoform detection  
compute consensus

report non redundant  
polished transcript sequences



alignment to genome  
(Minimap2, GraphMap2, GMAP...)

report genes/isoforms  
quantify

adapted from Gordon et al. 2015

# Isoform detection: PacBio's Iso-Seq3 + ToFU/Cupcake

<https://github.com/yลิปacbio/IsoSeq3/>

- ❑ will tend to **merge alternative transcripts** (heavily depends on the reference quality)
- ❑ computationally expensive
- ❑ tailored to **Pacbio reads only**
- ❑ scripts for exon-junction description and quantification



# Alternative isoforms detection pipelines

## Specialized for Pacbio

- ❑ SQANTI (reference genome, gff)
- ❑ ToFu (reference genome & limited *de novo*)
- ❑ TAPIS (reference genome)
- ❑ IsoCon (*de novo* correction and detection of different transcripts at the base level, targeted data)

## Specialized for Nanopore

- ❑ FLAIR (reference genome)

## Technology agnostic

- ❑ TALON (input = alignments to ref)
- ❑ MANDALORION
- ❑ TrackCluster (*de novo*)

# Pipelines focused on quantification

- ❑ Developed by Nanopore (based on alignment + Salmon)  
<https://github.com/nanoporetech/pipeline-transcriptome-de>
- ❑ LIQA (truncated reads treated using an EM algorithm)

# Application example



[Front Genet](#), 2021; 12: 683408.

PMCID: PMC8321248

Published online 2021 Jul 15. doi: [10.3389/fgene.2021.683408](https://doi.org/10.3389/fgene.2021.683408)

PMID: [34335690](https://pubmed.ncbi.nlm.nih.gov/34335690/)

## **PacBio Iso-Seq Improves the Rainbow Trout Genome Annotation and Identifies Alternative Splicing Associated With Economically Important Phenotypes**

[Ali Ali](#)<sup>1</sup>, [Gary H. Thorgaard](#)<sup>2</sup> and [Mohamed Salem](#)<sup>1,\*</sup>

### Long-read cDNA sequencing identifies functional pseudogenes in the human transcriptome

[Robin-Lee Troskie](#), [Yohaann Jafrani](#), [Tim R. Mercer](#), [Adam D. Ewing](#) ✉, [Geoffrey J. Faulkner](#) ✉ & [Seth W. Cheetham](#) ✉

[Genome Biology](#) **22**, Article number: 146 (2021) | [Cite this article](#)

2795 Accesses | 2 Citations | 31 Altmetric | [Metrics](#)

# Long reads miscellaneous

- Specific spliced alignment tools start to emerge (uLTRA, Sahlin et al. 2021)
  - Cleaning for spliced sites (with ref) TranscriptClean , FLAIR
  - Reference-free correction might become a standard in the years to come (isONcorrect, Sahlin et al. 2021) (!\ generally, do not use reference free correction methods tailored for genomic long reads)
  - Assembly using short+long reads+ref: StringTie2
  - De novo assembly (RNA-Bloom, Nip al. 2023)
- 
- A website that lists long reads tools: <https://long-read-tools.org/table.html>

# Next challenges with long reads

- ❑ guarantee full-length RNA or cDNA libraries
- ❑ sequence all different RNAs (not only poly-A)
- ❑ allele-specific transcripts
- ❑ acquire knowledge about 3' and 5' ends, polyA tails (homopolymers)
- ❑ new steps toward full de novo pipelines

# What was not viewed during this session

- bacterial RNA
- genome-guided assembly
- metatranscriptomics
- single cell RNA
- tools specialized for ncRNAs, smallRNAs
- tools specialized for fusion transcripts
- transcript annotation (<https://busco.ezlab.org/> for instance)
- ...
- up next**: differential study (statistics for RNA-seq)